Karlsruher Institut für Technologie

ETP-KA/2020-11

Development and Deployment of a Deep Neural Network based Flavor Tagger for Belle II

Jochen Gemmler

Zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

von der KIT-Fakultät für Physik des
Karlsruher Instituts für Technologie (KIT)

genehmigte

Dissertation

von

M.Sc. Jochen Gemmler

aus Ulm

Tag der mündlichen Prüfung:	15. Mai 2020
Referent:	Prof. Dr. Michael Feindt
Korreferent:	Prof. Dr. Florian Bernlochner

Contents

1. Introduction

The world around us is an amazing place. The richness of nature can be described to a high-level of accuracy by physics laws. In numerous high energy experiments, these theories are tested at the smallest scales. Modern experiments increasingly rely on advances in computing and analysis technology to pursue their goal. In this thesis, new machine learning tools designed for the Belle II experiment are presented and validated against Belle data.

The Standard Model of particle physics is a very successful attempt to describe fundamental particles and their interactions. It contains 6 different types of quarks, 6 different types of leptons, also referred to as flavors, and three of the four known fundamental forces. With the discovery of the Higgs boson in 2012, the last missing piece of the Standard Model (SM) has been found. Observations from high energy experiments are compatible with the SM with unprecedented precision. Nevertheless, there are many observations in physics which cannot be explained by the SM, for instance candidates for dark matter or dark energy. Many new theories predict small deviations from SM observables which can be tested by higher precision measurements. Therefore experiments are essential for the extension and improvements of models for the description of nature.

The Belle experiment and its successor, the Belle II experiment, are located at the Japanese High Energy Accelerator Research Organization (KEK) at a lepton collider with asymmetric beam energies. Both experiments are dedicated to high precision measurements of SM parameters which are sensitive to CP violation in the flavor sector. They are mainly designed to investigate B meson decays, which are decay products of the $\Upsilon(4S)$ resonance. This resonance decays into two charged or two neutral B mesons in almost all cases. Since the neutral B meson pairs originate from the same source, they are entangled by quantum mechanical principles. To be able to measure parameters of CP violation, the flavor of both B mesons has to be known at the time of their decays.

In order to build on the success of its predecessor, the Belle II experiment did not only set the focus on improvements of hardware but also in software development. The Belle II collaboration provides a software framework for recording, processing and reconstruction of particle decays on measured data. Algorithms which have proven to be useful over the last decade are revised and redeveloped in a more efficient or more precise manner.

An opportunity during this step is the integration of new paradigms to existing methods, for instance in the field of machine learning. New advances in data-driven methods have revolutionized the field. With the availability of improved hardware technologies, algorithms with a high number of adjustable parameters can be deployed. The so-called Deep Learning methods achieve a better representation of the investigated data, allowing a more accurate assignment of classes or regression of a function.

In this thesis a flavor tagging algorithm based on these principles is developed and investigated. It is essential for the usage of this algorithm to not only evaluate its performance on generated Monte Carlo data, but also its actual performance on experimental data. Potential differences have to be investigated and quantified to shed light on the accuracy and reliability of the algorithm. The algorithm is designed for the Belle II experiment, and is validated on existing Belle data. The effective tagging efficiency of the developed algorithm is determined on a selection of reconstructed $B^0 \to D^{*-}\pi^+$ and $B^- \to D^0\pi^-$ decays and compared to the efficiency of an alternative, more traditional approach. Although the chosen hadronic decays have lower statistics than semi-leptonic decays, the reconstruction is much cleaner, and a lower cross-feed between signal side and tag side is expected.

The Belle and Belle II experiments are described in Chapter 2. The physics background of the analysis is outlined in Chapter 3. The majority of this thesis focuses on the development and calibration of a flavor tagging algorithm, which is described in Chapter 4. The calibration procedure is presented in Chapter 5. Here, for the reconstruction of events the Belle II analysis software framework is used. In Chapter 6 a combination of multiple flavor tagging algorithms is studied. First applications on Belle II data are shown in Chapter 7. An additional application for machine learning is investigated in Chapter 8, where the utilization of generative adversarial neural networks for a speed up of the simulation of energy depositions in calorimeters is investigated.

2. The Belle and Belle II Experiments

The Belle II experiment is a particle detection experiment located at SuperKEKB, an asymmetric e^+e^- collider in Tsukuba, Japan. The detector is installed at the High Energy Accelerator Research Organization (KEK) and has replaced its predecessor, the Belle experiment. Both experiments are mainly designed to investigate B meson decays, which are produced by decays of the $\Upsilon(4S)$ resonance with the energy $E_{e^+e^-} = 10.58$ GeV in the center of mass frame of the colliding e^+e^--pair.

While writing this thesis, the Belle II experiment was in the starting phase and the data set recorded by the Belle experiment was converted to be readable and useable by the Belle II software framework. This makes it possible to test algorithms which are developed for Belle II directly on Belle data, enabling a more data driven approach for algorithm development.

In this chapter a brief description of the experimental setup of both detectors will be provided – in Section 2.1 a summary of the experimental setup of Belle is shown, while Section 2.2 focuses on the changes of Belle II.

2.1. The Belle Experiment

This section intends to give a brief overview of the sub-detectors of Belle and the KEKB accelerator. A more detailed description can be found in [1, 2], which are main sources for this section. Belle recorded data for more than a decade. In total 988.3 fb^{-1} of integrated luminosity was recorded, including 711 fb^{-1} at the $\Upsilon(4S)$ resonance. Together with the BaBar experiment, which was located at the Stanford Linear Accelerator Center (SLAC), it provided unique insights into the Yukawa sector.

2.1.1. KEKB

The KEKB storage ring consisted of two separate beam pipes which crossed each other at the interaction point. The storage ring operated at asymmetric energies

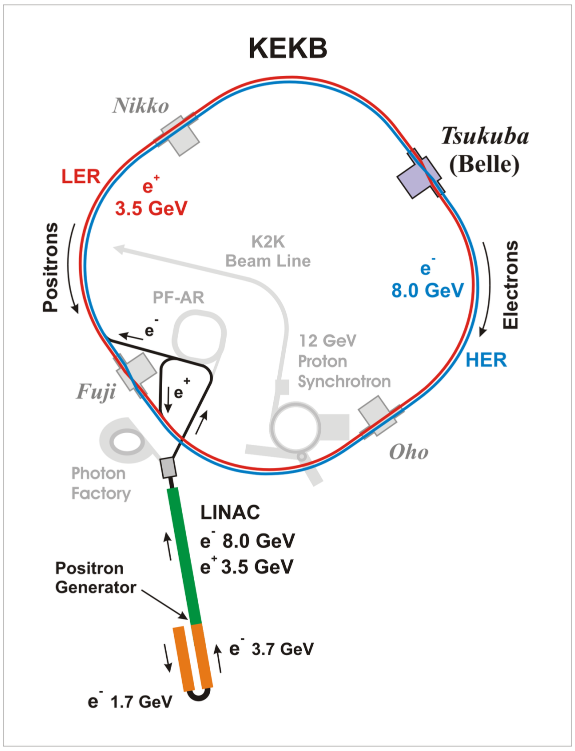

Figure 2.1.: Schematic view of the KEKB storage ring in Tsukuba, Japan. It consists of two separated beam pipes, with a low energy beam (LER) and a high energy ring (HER). The Belle experiment was located at one of the two beam crossing points. The ring is now upgraded to SuperKEKB and the Belle detector replaced with the Belle II detector. Taken from [3].

of 8 GeV for electrons and 3.5 GeV for positrons, which were injected into a linear accelerator (linac). The asymmetric beam energies caused a boost of the interaction particles with a Lorentz factor of $\beta\gamma = 0.425$. Due to radiation losses, there had to be a continuous injection of particles into the storage rings. The peak luminosity at the accelerator was $2.1 \cdot 10^{34}$ cm^{-2}s^{-1}. The beam size at the interaction point was 80 µm.

2.1.2. The Belle Detector

The Belle detector was primarily designed to investigate decays of the $\Upsilon(4S)$ resonance into two B mesons. It was composed of several sub-detectors, to measure specific properties of the resulting decay products, e.g. momentum, the ionization energy loss per distance unit $\mathrm{d}E/\mathrm{d}x$, charge or the vertices of sub-decays.

The detector consists of a silicon vertex detector (SVD), a central drift chamber (CDC), an electromagnetic calorimeter (ECL), and a hadron calorimeter (KLM). To identify for a certain particle types the aerogel Cherenkov counter (ACC), a time-of-flight (TOF) system and and extreme forward calorimeter (EFC) are particularly helpful. The inner parts of the detector were immersed in a homogenous electromagnetic field with field strength of 1.5 T. A schematic overview over the most relevant detector parts is shown in Figure 2.2.

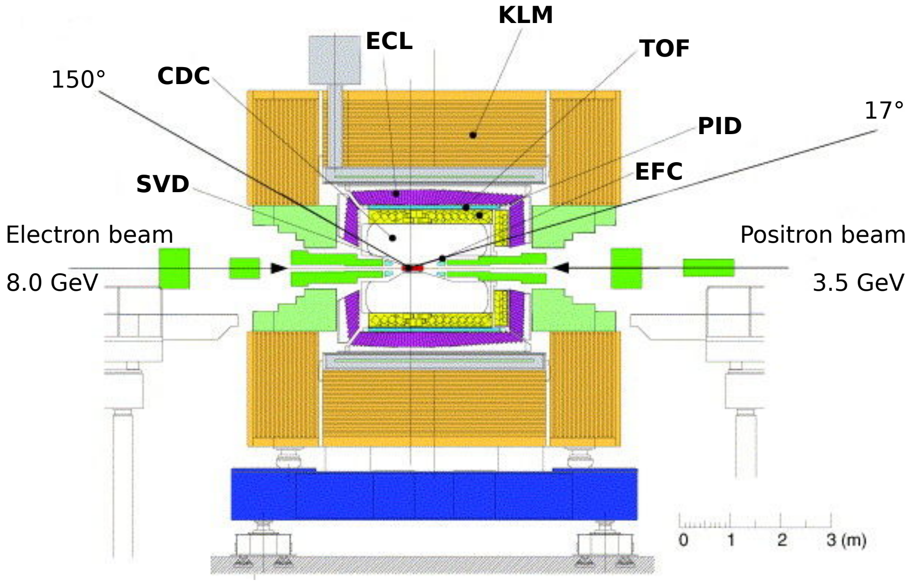

Figure 2.2.: Schematic view of the Belle detector. Adapted from [1].

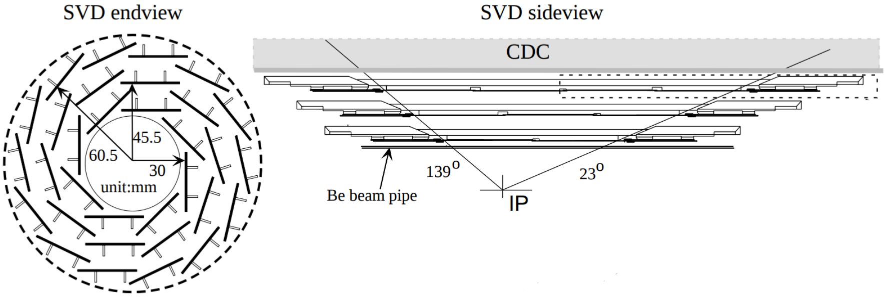

Figure 2.3.: Schematic representation of the SVD. Adapted from [1].

The Silicon Vertex Detector

The inner-most layer of the Belle detector was a silicon strip detector. To be able to precisely determine vertices from reconstructed tracks, a high resolution of the **silicon vertex detector** was required close to the interaction point.

The SVD was upgraded in several steps during the lifetime of Belle. The final version was called SVD2 and offered an acceptance of $17^\circ < \theta < 150^\circ$. Here θ denotes the angle with respect to the beam axis. The majority of data, 85%, was recorded with this configuration. It is the closest sub-detector to the beam pipe with an inner radius of only 15 mm with respect to the interaction point and therefore very susceptible to beam-induced backgrounds caused by interactions of the electron and positron beams. At peak luminosity of KEKB, the occupancy of the innermost layer was in the range of 5-7%.

The Central Drift Chamber

The **Belle Central Drift Chamber** was filled with helium and ethane in equivalent ratio. The radius of the CDC ranges from 80 mm to 880 mm. It was composed of 50 cylindrical layers, each of which contains a multitude of sub-layers.

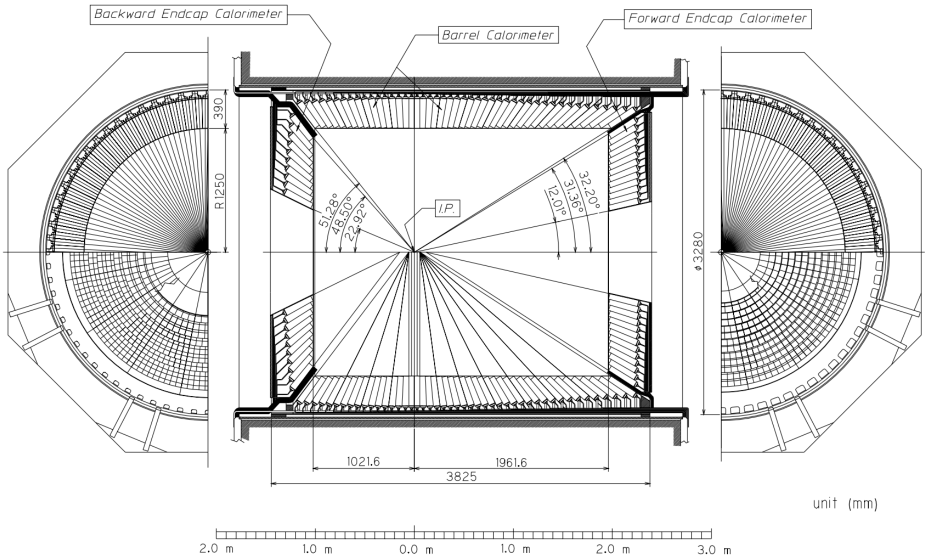

Figure 2.4.: Schematic view of the ECL. It is divided into barrel and end-cap regions. Also clearly recognizable is the acceptance of the ECL. Most components of the calorimeter have been re-used for Belle II. Taken from [1].

The electromagnetic field causes charged particles to propagate in a helix-like trajectory which allows the determination of their charge and momentum. A common interaction with the detector material is the so-called multiple Coulomb scattering, which forces the reconstruction algorithms to re-parametrize the reconstructed trajectories. This issue is increasingly important for low momentum tracks. The CDC also contributes to particle identification by measuring $\mathrm{d}E/\mathrm{d}x$ of charged particles.

The Electromagnetic Calorimeter

The main task of the electromagnetic calorimeter (ECL) is the detection of photons, e.g. from $\pi^0 \to \gamma\gamma$ decays, charged and neutral particle discrimination and assists in luminosity measurements. Traversing charged particles cause photon showers in the scintillator material, which are captured by silicon photo diodes. By measuring the energy of those showers in the ECL, the E/p ratio can be determined. Here E denotes the recorded and calibrated energies in the ECL and p the reconstructed momentum, using the dedicated Belle tracking detectors. The E/p ratio allows for the discrimination of hadrons from electrons and muons. The main constituents of the ECL are 8736 CsI(Tl) crystals. It is sub-divided into different parts, the barrel and the end-cap regions. The crystals have a slanted design and varying geometries. A schematic illustration is shown in Figure 2.4. Additionally, an extreme forward calorimeter (EFC) composed of Bismuth Germanate Oxide directly on top of the final quadrupole magnets was used to monitor beam background.

The muon and K_L^0 detector

In the outermost parts, the muon and K_L^0 detector system (KLM) was installed. It was an alternating setup of iron absorber plates and double-gap resistive plate counters (RPC). The RPC modules contained alternating layers with insulators, gas gaps and a non-flammable gas mixture of HFC-134a, Argon and Butane-Silver. Signals of the KLM allowed the reconstruction of K_L^0 particles and muons.

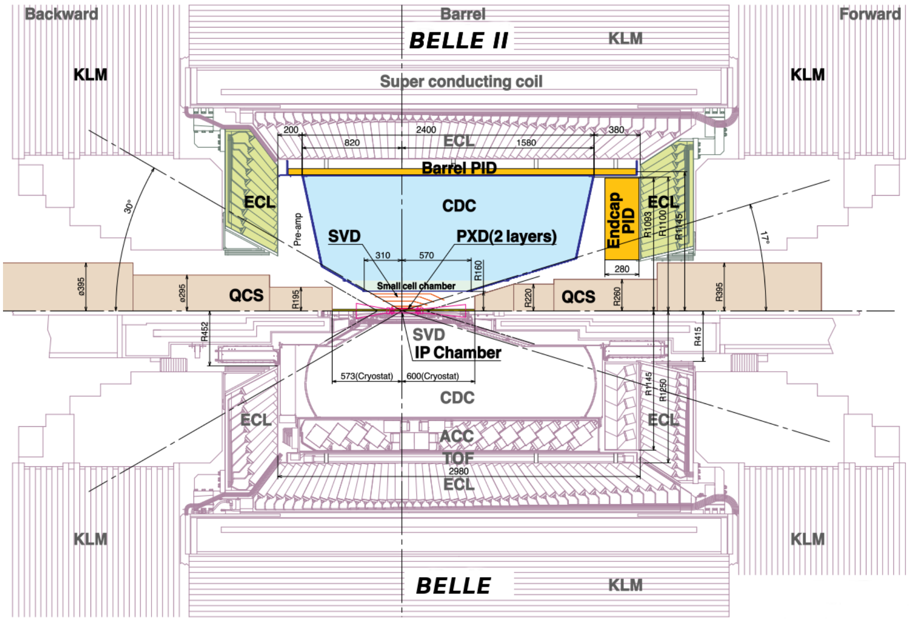

Figure 2.5.: Schematic view of the upgraded Belle II detector (top half) in comparison to the Belle detector (bottom half). Adapted from [4].

Particle Identification

For the discrimination between particle types, the information of various sub-detectors is combined. The **time-of-flight system** (TOF) consisted of multiple scintillator and photomultiplier modules. It was located in the barrel region. With its total of 64 modules it was mainly designed for kaon-pion discrimination.

The **aerogel Cherenkov counters** (ACC) were located in the barrel and end-cap regions. The modules of the ACC contained ten different types of refractive material, varying in the range of $n = 1.01, ..., 1.03$. The Cherenkov light, produced by charged particles which were traveling with a velocity above the phase speed of light in the medium, was captured by photomultipliers of varying diameter.

2.2. The Belle II Experiment

Conceptually, the Belle II experiment is similar to its predecessor in many ways and some detector components, like parts of the CDC, have even been reused. But significant parts have changed. These changes will be elaborated on in the following section. A schematic view comparing the Belle and the Belle II detector is shown in Figure 2.5. The main source of technical details in this section is the Technical Design Report [4], which contains a more detailed overview of the Belle II detector.

The storage ring has been upgraded to **SuperKEKB**. This implies asymmetric beams are readjusted and a nano beam scheme is implemented, which allows the beam to be collimated at the interaction point with a vertical beam size in the nanometer regime. The boost factor has been reduced significantly to $\beta\gamma = 0.28$, since the collider now operates at 7 GeV/c and 4 GeV/c for the high energy and low energy electron beams. In combination with a higher beam current, its projected peak luminosity can be increased by a factor of 40, while an increased amount of beam background is also expected. The main beam background processes are beam gas scattering with residual gas atoms and Touscheck radiation, which is caused by Coulomb scattering of the stored particles.

The most important upgrades are:

- A new **pixel detector** as the innermost layer has been introduced. While it does not support stand-alone tracking, is intended to increase the resolution of reconstructed tracks in a high-occupancy environment. The projected occupancy of the PXD is is expected to be in the range of $\sim 1\%$.
- The **silicon vertex detector** has been completely rebuilt. A precise determination of track and impact parameters near the beam spot to improve vertexing is crucial for Belle II to reach its physics goals. These goals include time-dependent measurements of CP violation, Flavor Tagging algorithms (Section 3.3.1) are essential. The number of layers has been increased to four with respect to the Belle design. In the forward region the detector has a slanted design to obtain a similar coverage to the Belle detector. In contrast to Belle, the Belle II SVD is capable of stand-alone tracking. Furthermore the design increases the resolution of low momentum tracks.
- Some of the **particle identification systems** have been replaced. Instead of TOF, a time-of-propagation (TOP) counter is used. The ACC has been replaced by an aerogel ring-imaging cherenkov counter (ARICH) in the forward and end-cap regions. A two layer aerogel configuration with different refractive indices allows a more accurate determination of particle velocity.

In 2018, after a commissioning phase, the Belle II experiment started to record first collisions. The commissioning phase was mostly used for alignment and efficiency studies for different sub-detectors. The following phase already achieved an integrated luminosity of $(496.3 \pm 0.3 \pm 3.0)$ pb^{-1} on the $\Upsilon(4S)$ resonance[5]. While many effects on data regarding the detector parts and background events still have to be understood, first physics results are already possible, taking the systematics into account[6]. The recording period for physics measurements started in March 2019 and contains a data already set with an integrated luminosity of 8.7 fb^{-1} on the $\Upsilon(4S)$ resonance and 0.8 fb^{-1} off-resonance. The projected luminosity is shown Figure 2.6 and it is intended to supersede the integrated Belle luminosity by a factor of 50 until 2027.

2.3. Software Tools at the Belle II Experiment

Not only the Belle II detector has been upgraded, but also the software tool kit of the Belle II experiment has been completely reconditioned. Many new and powerful software libraries have been developed and many modern concepts in software design have been introduced. Multivariate methods and application tools are of more relevance today and the usage is increasingly and successfully explored in high energy physics [8, 9] While in high energy physics `C++` and the `ROOT` libraries [10] are still playing a major role, many steps of the data analysis have been migrated to `Python` tool kits for data visualization and rapid prototyping. Thus, some of the major improvements of Belle II do not only arise from a better and more modern detector, but in software, which helps the experiment improve its data processing at a lower cost and to achieve better physics measurements.

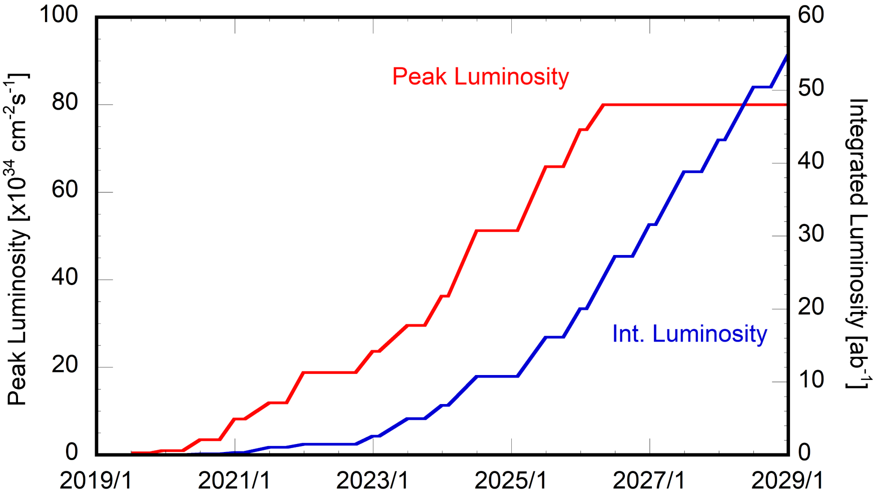

Figure 2.6.: The projected long-term luminosity of SuperKEKB. Taken from [7].

2.3.1. The Belle II Analysis Software

The Belle II analysis software framework (BASF2)[11, 12] is a software framework which is used in data acquisition, tracking, reconstruction and analysis. The software is used for online analysis, during data taking, as well as offline analysis. For offline analysis, BASF2 can be executed on the LHC Computing Grid (LCG) or on single desktop machines, with the support of multi-processing. It therefore allows a flexible rapid development of an analysis and a fast trivial parallelization for large scale computing, which can be described by Amdahl's law

$$S(s,p) = \frac{1}{1-p+\frac{p}{s}} \tag{2.1}$$

where S is the speedup, p is the fraction of time spend in the parallelizable part and s in the non-parallelizable (serial) part. While runtime dependent tasks are written in C++, C and Fortran, the user interface is based on Python scripts. The internal interface for that is provided by the boost library.

One of the key aspects of BASF2 is a modular design and a direct access to a data store. The user can build directed acyclic graphs via the Python scripts, so-called path objects, which chain different modules. The configuration of these modules has to be provided by the user in these scripts. Once a path object is created, it can be executed and processed. These tasks can be very different and range from a mere Monte Carlo generation with evtgen, pythia, detector simulation with GEANT4 [13] or the reconstruction of user specific decay channels. The information is saved with a format convention in so-called mini data summary tables (mDST). This includes only a small fraction of the information available after reconstruction and processing. For instance helix parameters and combined likelihood ratios from tracking are included. Once these channels are reconstructed, the user can write certain variables, which have to be defined in the VariableManger class into C++, to flat tuples. The ROOT data format currently is the default format of the experiment. A support of the HDF5 data format is in discussion.

BASF2 also provides helpful modules for the execution of certain tasks. These modules are required in many different analyses, e.g. the suppression of continuum

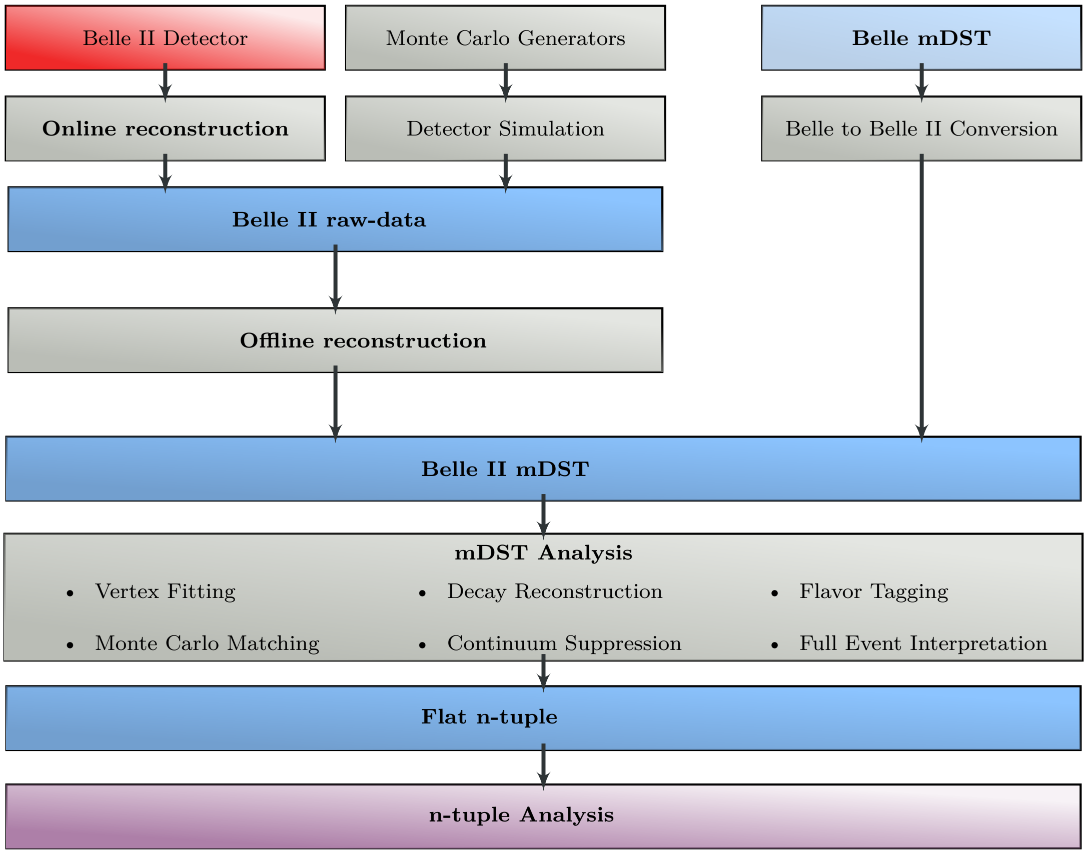

Figure 2.7.: Workflow of a data processing with the BASF2 framework. Adapted from [14].

(Section 2.3.2), tagging algorithms or for the reconstruction of a complete event(see Section 2.3.3).

A package dedicated for interfacing multivariate analyses was created by [14], the MVA package. It supports common packages, e.g. TMVA, or fastbdt, a cache friendly gradient boosted decision tree implementation[15]. Major contributions for accessing and using arbitrary modern machine learning frameworks (e.g. tensorflow) via a general Python interface and to Belle II analysis software framework have been made in the scope this thesis.

The b2bii package [16] provides the necessary tools to convert Belle MC and data, which are stored in PANTHER based storage tables. Helix parameters and other quantities are converted, in order to fit with the mDST data format of BASF2. Here, contributions have been made, as well.

A scheme of the typical data processing flow in an analysis with the BASF2 software framework is visualized in Figure 2.7.

2.3.2. Continuum Suppression

The continuum suppression module in BASF2 provides an implementation of a high variety of discrimination variables which carry information, whether an event originates from an $\Upsilon(4S)$ resonance or if it has a background like signature. These variables are used in a high number of analyses and are a good example of the usefulness of

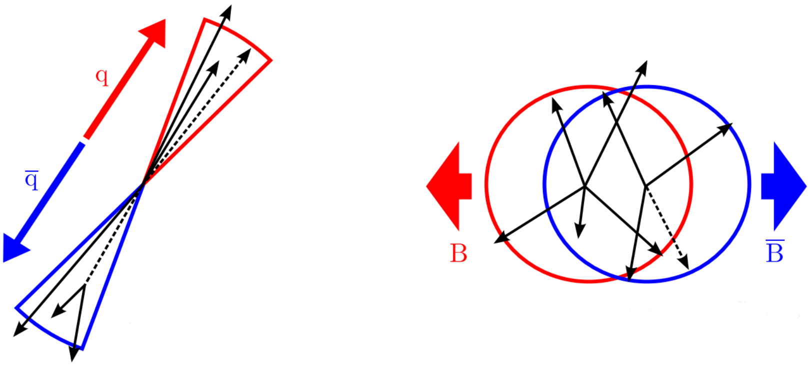

Figure 2.8.: Illustration of a background-like (left) and a B meson-like (right) decay event. Decays of the $\Upsilon(4S)$ resonance have a more spherical signature. Adapted from [17].

pre-implemented specialized modules. In the following section, the most relevant aspects for this thesis are discussed. More details can be found in [2, 17].

One of the main background components originates from

$$e^+e^- \to q\bar{q} \tag{2.2}$$

processes where no $\Upsilon(4S)$ resonance, but a quark–anti-quark pair (u, d, c, s) is produced in an annihilation process. Those events have a distinct topological signature. Background events have a more jet like structure. The hadrons are produced in back-to-back direction in the center of mass frame. B meson decays on the other hand show a more isotropic distribution. They have a very low momentum in the $\Upsilon(4S)$ frame, due to the low mass difference between the mother and daughter particles of the decay process. The signature for the different event types, including a schematic of the momenta are, illustrated in Figure 2.8.

To exploit this topological information, the thrust T_{th} for an event with N particles with momenta $\boldsymbol{p_i}$ is defined

$$T_{\text{th}} = \max_{\boldsymbol{T}} \frac{\sum_{i=1}^{N} |\boldsymbol{T} \cdot \boldsymbol{p_i}|}{\sum_{i=1}^{N} |\boldsymbol{p_i}|}, \tag{2.3}$$

where the maximization with respect to the axis T. The axis for which the ratio is maximal is called the thrust axis $\boldsymbol{T_{\text{th}}}$. For events which decay perfectly back-to-back the thrust $T_{\text{th}} = 1$ is the maximal value, whereas for spherical events, the quantity is in the regime of $T_{\text{th}} \sim 0.5$.

If one B meson is already reconstructed, this information can be used for background discrimination. The thrust axis can be defined only for the remaining particles which do not belong to the reconstructed B meson, the so-called rest of event.

A useful definition is the angle between the thrust axis of the reconstructed B meson and the thrust axis of the rest of event $\cos(\theta_{\text{th}})$.

Additional important quantities are the **Fox-Wolfram moments** H_k. Here, the spherical topology of a signal event is used, but no distinct thrust axis is defined.

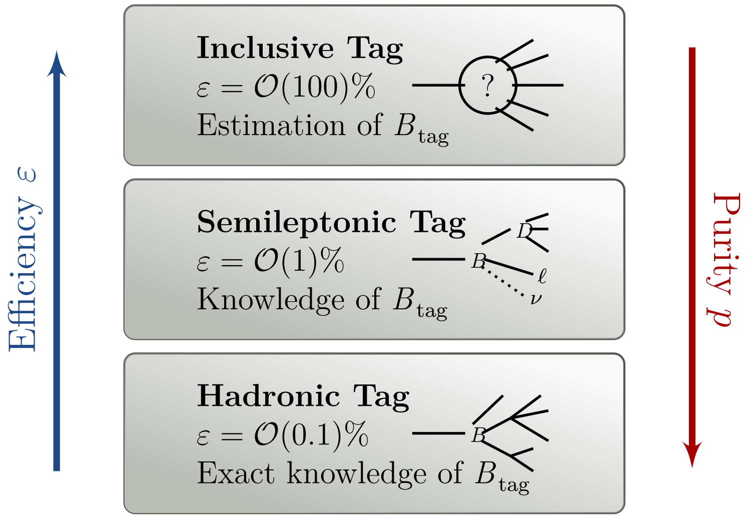
Inclusive Tag
$\varepsilon = \mathcal{O}(100)\%$
Estimation of B_{tag}
?
Semileptonic Tag
$\varepsilon = \mathcal{O}(1)\%$
Knowledge of B_{tag}
B
D
ℓ
ν
Hadronic Tag
$\varepsilon = \mathcal{O}(0.1)\%$
Exact knowledge of B_{tag}
B
Efficiency ε
Purity p

Figure 2.9.: Overview of different tagging procedures to categorize B mesons in $\Upsilon(4S)$ decays. Adapted from [14].

Rather a basis of rotationally invariant observables is chosen [18]. The event is parametrized in Spherical Harmonics

$$H_k = \sum_{i,j}^{N} |\boldsymbol{p_i}||\boldsymbol{p_j}|P_k(\cos\theta_{ij}), \tag{2.4}$$

where P_k are the Legendre polynomials and θ_{ij} denotes the relative angle between the different particle momenta. The lower Fox-Wolfram moments are the most relevant. The ratios

$$R_k = \frac{H_k}{H_0} \tag{2.5}$$

provide a clear signature for collimated events. R_k can take values in the range of 0 or 1 for odd or even elements k, respectively.

2.3.3. Tagging Methods

The automated classification of B meson decays is crucial in most Belle and Belle II analyses. Depending on the type of the analysis, one can attempt to associate a specific decay chain, a certain decay product, or a distinct flavor to one or both B mesons. This procedure is called **tagging** – the established tagging algorithms can be summarized in three distinct categories:

- Hadronic tagging,
- semi-leptonic tagging and
- inclusive tagging.

The efficiency and purity determine the quality of the tagging procedure. The purity is defined by the number of true signal events compared to the number of reconstructed events. The efficiency in this case is the fraction of events, that can be tagged with respect to all $\Upsilon(4S)$ decays. Both quantities allow for a reasonable decision of which algorithm is best suited for a certain type of measurement.

For **hadronic tagging** the decay chain of a B meson decay is completely reconstructed – all the final states of the decay chain are principally measurable at Belle or Belle II. This allows for an easier kinematical selection of the decay channels. Since hadronic decays are relatively clean, a reconstruction with a high purity is possible. Unfortunately, low branching fractions of hadronic B meson decays of the order $\mathcal{O}(10^{-3})$ result in low efficiencies.

In **semi-leptonic tagging** decays are only partially reconstructed. In these decays the b quark decays into a lighter quark mediated by a W boson which can couple to a lepton and neutrino pair. The neutrino escapes the detector unseen. In this exclusive tagging procedure, the efficiency is higher by a factor of 10 compared to the hadronic case.

In the **inclusive** case, the decay is not explicitly reconstructed, but only an estimation if a specific decay has occurred, is provided. Here, decay specific attributes e.g. the momentum $\boldsymbol{p}_B$ of the B meson is approximated. While this allows an efficiency up to the order of $\mathcal{O}(1)$, the purity is usually lower. An overview of the different tagging mechanisms is provided in Figure 2.9.

At Belle II the algorithm for exclusive tagging of an event is referred to as the Full Event Interpretation[19]. An example for inclusive tagging is Flavor Tagging. Here, the B meson flavor of an event is predicted by a multivariate classifier (Chapter 4) on the basis of the detected decay products. The algorithm and its foundations are described in the next chapter.

3. Theoretical Background

The Standard Model of particle physics (SM) describes physics at the shortest distance scales probed so far. The SM provides the theoretical foundations for the analysis in this thesis. The salient features of the SM are summarized in this chapter. Further details can be found in many references, e.g. [20, 21].

3.1. The Standard Model

The SM describes the fundamental forces, except the gravitational force, and particles of the universe to the best of our knowledge. The particles can be subdivided in leptons, quarks and gauge bosons. The forces are mediated via the gauge bosons, including the

- weak force – two charged ($W^{\pm}$) and a neutral (Z) bosons,
- strong force – 8 gluons with color charge,
- electromagnetic force – one massless photon without electric charge.

The gravitational force is not incorporated into the SM. The SM can be described in terms of symmetry groups by a combination of unitary and special unitary groups. The fields can be parametrized by operators, following a

$$SU(3)_C \times SU(2)_L \times U(1)_Y \tag{3.1}$$

gauge symmetry, where the strong interaction obeys a $SU(3)_C$ symmetry and the electroweak interaction can be described via the $SU(2)_L \times U(1)_Y$ group. It is spontaneously broken in the vicinity of the vacuum energy by the Higgs field, resulting in a $U(1)_{EM}$ symmetry.

With the Brout-Englert-Higgs mechanism, a theory has been introduces that describes the breaking of the continuous symmetry and explains the mass of fundamental particles. The spontaneous breaking leads to Goldstone bosons, three of them are “absorbed” by the corresponding gauge bosons, giving them a mass.

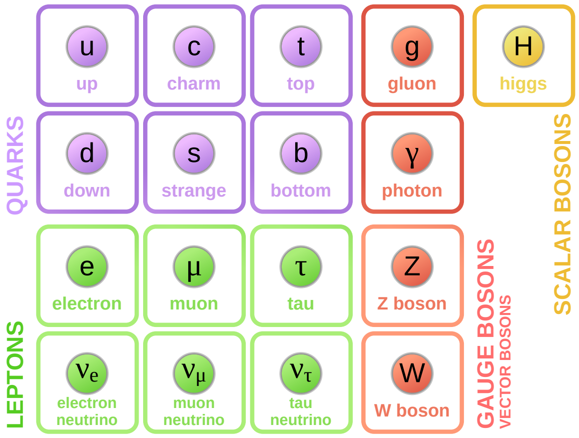

Figure 3.1.: Elementary Particles of the Standard Model of Particle Physics. Adapted from [22].

The SM has precisely measured parameters over a large set of different scales. The Higgs boson was discovered in 2012 at the LHC as the last missing piece of the SM [23, 24].

At the time of writing, no evident discrepancy with the SM has been found. However, there are statistical deviations of measurements from the predicted observables, so-called tensions. They will be described further in Section 3.2.

Nevertheless, the SM is not yet sufficient to explain many observations. Cosmological observations and rotation curves of galaxies indicate that a form of dark matter and dark energy is present. Theories which have the power to potentially unify many of these observations, e.g. supersymmetry, have been proposed. Supersymmetry introduces a super partner for every particle in the SM. This can explain the hierarchy between the weak scale and the Planck scale naturally. However, the parameter space of the model is vast, making it currently impossible to scan over the entire range.

Hence, it is important to continue probing the SM at even higher precision in order to obtain insights as to how the model can be improved, or other ideas have to be explored. While there are a number of different concepts to explain observations which imply the existence of dark matter, dark energy or similar, measurements remain key to examine the (in)validity of such concepts.

The flavor sector, see Section 3.2, allows to probe for a range of interesting quantities like CP violation, possible dark matter candidates and high precision measurements of fundamental parameters of the SM.

3.2. Flavor Physics at Belle and Belle II

Flavor physics is one of the most interesting topical areas in high energy physics. New experiments like the Belle II experiment are evolving. There are tensions between current measurements and the SM which as in to shed light on a $\mathcal{R}(D^{(*)})$ measurement [25], where lepton flavor universality is being challenged. In these measurements, ratios of branching fractions of exclusive semileptonic B decays to the tau lepton versus the lighter lighter lepton flavors e^- and μ^- are investigated. Higher precision and a better understanding of underlying systematic effects might improve measurements further in the future. Especially CP violation provides a unique playground for testing and probing the SM. The following section highlights important aspects of flavor physics with a focus on the properties B meson of decays.

3.2.1. CKM Matrix

In the SM, there are three families of quarks; each of them consists of an up-type and a down-type quark. Except for the mass, up-type and down-type quarks have similar attributes, e.g. charge and spin. Also the coupling behavior of each family is similar. Each type of quarks is assigned a unique flavor:

$$\begin{pmatrix} \text{up} \\ \text{down} \end{pmatrix}, \begin{pmatrix} \text{charm} \\ \text{strange} \end{pmatrix}, \begin{pmatrix} \text{top} \\ \text{bottom} \end{pmatrix}. \tag{3.2}$$

The symmetry of the SM allows for neutral and charged interactions between fermions and gauge bosons. Charged currents can violate flavor linking down-type weak eigenstates (d', s', b') to down-type mass eigenstates (d, s, b).

This relation between the weak and mass eigenstates of quarks was initially proposed by Nicola Cabibbo. Extended to the third generation with the Cabibbo-Kobayashi-Maskawa (CKM) mechanism [26], the CKM matrix

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix} \tag{3.3}$$

describes the relation between the two bases. In the SM, all flavor violation proceeds through the CKM matrix. At tree-level, the CKM matrix only appears in charged current processes. The CKM matrix is a product of two unitary transformations in the SM, and hence is predicted to be unitary. The measurements of the CKM matrix are already very precise [27]

$$|V_{\mathrm{CKM}}| = \begin{pmatrix} 0.97446 \pm 0.00010 & 0.22452 \pm 0.00044 & 0.00365 \pm 0.00012 \\ 0.22438 \pm 0.00044 & 0.97359^{+0.00010}_{-0.00011} & 0.04214 \pm 0.00076 \\ 0.00896^{+0.00024}_{-0.00023} & 0.04133 \pm 0.00074 & 0.999105 \pm 0.000032 \end{pmatrix}. \tag{3.4}$$

Since experimentally obtained values also satisfy the unitarity condition, this is one of many strong hints, that three generations of quarks are sufficient to describe phenomenology of the related physical processes.

The CKM matrix can be parametrized by three mixing angles and a complex phase. A convenient way to parametrize it, is the Wolfenstein parametrization [28], λ, A, $\bar{\rho}$ and $\bar{\eta}$, where

$$\lambda = \frac{|V_{us}|}{\sqrt{|V_{ud}|^2 + |V_{us}|^2}}, A\lambda^2 = \lambda \left| \frac{V_{cb}}{V_{us}} \right|, A\lambda^3(\rho + i\eta) = V_{ub}^*. \tag{3.5}$$

In the Wolfenstein parametrization up to the fourth order in λ,

$$V_{\mathrm{CKM}} = \begin{pmatrix} 1 - \lambda^2/2 & \lambda & A\lambda^3(\rho - i\eta) \\ -\lambda & 1 - \lambda^2/2 & A\lambda^2 \\ A\lambda^3(1 - \rho - i\eta) & -A\lambda^2 & 1 \end{pmatrix} + \mathcal{O}(\lambda^4), \tag{3.6}$$

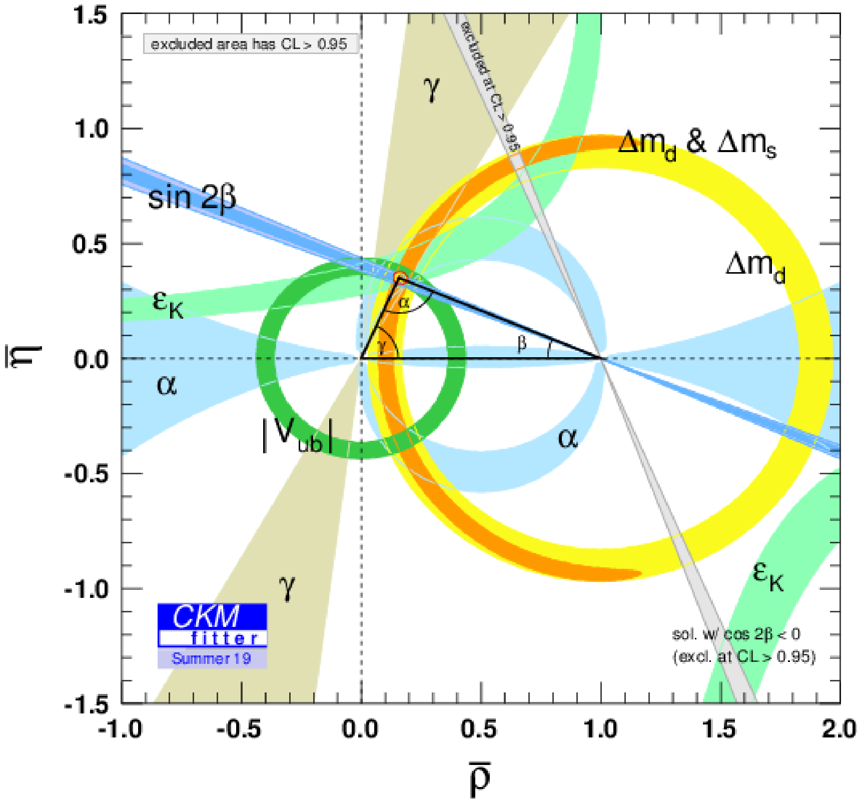

Figure 3.2.: Constraints in the $(\bar{\rho}, \bar{\eta})$ plane. Taken from the CKMfitter group[29]. EPS 2019 conference.

where the diagonal like structure is clearly visible. At all orders of the small parameter $\lambda \approx 0.22$ the CKM matrix is unitary. The unitarity conditions can be visualized as six different triangles in the complex plane. For visualization and measurements, usually the unitarity condition

$$V_{ud}V_{ub}^* + V_{cd}V_{cb}^* + V_{td}V_{tb}^* = 0 \tag{3.7}$$

is used and normalized on the $V_{cd}V_{cb}^*$ side.

In order to improve the accuracy, $\mathcal{O}(\lambda^5)$ corrections to V_{td} are included with [30] $V_{td} = A\lambda^3(1 - \bar{\rho} - i\bar{\eta})$, where $\bar{\rho} = \rho(1 - \lambda^2/2)$ and $\bar{\eta} = \eta(1 - \lambda^2/2)$. In the complex $(\bar{\rho}, \bar{\eta})$ plane, the corresponding triangle is referred to as the "unitarity triangle".

The current state of measurements of this triangle is visualized by the CKMfitter group in Figure 3.2. The plot also includes constraints from flavor observables.

The phases of the unitarity triangle are given by

$$\beta = \phi_1 = \arg\left(-\frac{V_{cd}V_{cb}^*}{V_{td}V_{tb}^*}\right), \alpha = \phi_2 = \arg\left(-\frac{V_{td}V_{tb}^*}{V_{ud}V_{ub}^*}\right), \gamma = \phi_3 = \arg\left(-\frac{V_{ud}V_{ub}^*}{V_{cd}V_{cb}^*}\right) \tag{3.8}$$

and can be obtained by measuring fractions of specific matrix elements.

3.2.2. Neutral *B* Meson Mixing

In the SM, the unitarity of the CKM matrix ensures relations as shown in Equation (3.7), which implies the absence of flavor changing neutral currents (FCNC) at tree level. This makes flavor physics, in particular neutral meson mixing an interesting candidate to search for new physics beyond the SM.

While in the neutral Kaon system, mixing was known since 1964 [31] and was well investigated, e.g. the Argus Collaboration measured mixing in the neutral B^0 $(\bar{b},d)$ meson system in 1987 [32]. It was also observed in B_s^0 $(\bar{b},s)$ and D^0 meson systems.

Since one of the main topics of this thesis are neutral B mesons, the next sections focus the properties of the B^0 meson system. The following section is based on [33], where more experimental details can be found.

The time evolution of a B^0 meson can be described by the time-dependent Schrödinger equation

$$i\frac{\partial}{\partial t}\left|B^0(t)\right\rangle = \mathcal{H}\left|B^0(t)\right\rangle, \tag{3.9}$$

with the interesting consequence that even for a pure flavor initial state, the wave function of an evolved state can contain a mixture of both B meson flavors. This can be formulated analogously for the $\bar{B}^0$ case.

One key aspect is that mass eigenstates differ from CP eigenstates. This leads to B^0-$\bar{B}^0$ oscillations, where the oscillation frequency is determined by the mass difference between the light B meson state B_L and the heavy meson state B_H. Both masses are in a similar regime and lifetimes differ only slightly. The rotation can be described by the complex parameters p, q, which are constrained to $|p|^2 + |q|^2 = 1$

due to the normalization,

$$|B_L\rangle = \frac{1}{\sqrt{2}}\left(p\,|B^0\rangle + q\,|\bar{B}^0\rangle\right) \tag{3.10}$$

$$|B_H\rangle = \frac{1}{\sqrt{2}}\left(p\,|B^0\rangle - q\,|\bar{B}^0\rangle\right). \tag{3.11}$$

B mesons are short-lived particles with a lifetime in the regime of picoseconds. The decay rate Γ can be calculated by Fermi's golden rule

$$\Gamma = 2\pi \sum_f \left|\langle f\,|\,\mathcal{H}_W\,|\,B^0\rangle\right|^2 \rho_f \tag{3.12}$$

by integrating over all final states f with the corresponding density of states ρ_f, where $\mathcal{H}_W$ denotes the interaction hamiltonian.

It can be approximated as a quantum mechanical two-state system. An effective Hamiltonian can be constructed, although because of the finite lifetime the effective Hamiltonian is not hermitian itself (e.g. it does not conserve energy).

The mass of two constituent quarks as well as the the QCD, EM, and weak interactions, including higher orders, contribute to the Hamiltonian.

$$\mathcal{H}_{\text{eff}} = \boldsymbol{M} - \frac{i}{2}\boldsymbol{\Gamma} = \begin{pmatrix} M_{11} & M_{12} \\ M_{12}^* & M_{22} \end{pmatrix} - \frac{i}{2}\begin{pmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{12}^* & \Gamma_{22}. \end{pmatrix} \tag{3.13}$$

The two components $\boldsymbol{M}$ and $\boldsymbol{\Gamma}$ are hermitian by themselves with the conditions $M_{12} = M_{21}^*$ and $\Gamma_{12} = \Gamma_{21}^*$. From the requirement of CPT invariance, the conditions

$$M_{11} = M_{22} = m \tag{3.14}$$

$$\Gamma_{11} = \Gamma_{22} = \Gamma \tag{3.15}$$

follow. In the mass eigenbasis, the time evolution can be calculated by solving the time-dependent Schrödinger equation.

$$|B_L(t)\rangle = e^{-i\omega_L t}\,|B_L\rangle \tag{3.16}$$

$$|B_H(t)\rangle = e^{-i\omega_H t}\,|B_H\rangle \tag{3.17}$$

Since the eigenvalues $\omega_{L,H} = m_{L,H} + i\frac{\Gamma_{L,H}}{2}$ are known in the mass eigenbasis it is useful to express the mass difference and the decay rate difference between B_H and B_L as

$$\Delta m = m_H - m_L \tag{3.18}$$

$$\Delta\Gamma = \Gamma_L - \Gamma_H \tag{3.19}$$

respectively.

The eigenvalues can be used to calculate the time evolution in the flavor eigenbasis

$$|B(t)\rangle = \frac{1}{2p}\left(|B_L(t)\rangle + |B_H(t)\rangle\right) \tag{3.20}$$

$$= g_+(t)\,|B\rangle + \frac{q}{p}g_-(t)\,|\bar{B}\rangle \tag{3.21}$$

with

$$g_{\pm}(t) = \frac{1}{2}\left(e^{-i\omega_L t} \pm e^{-i\omega_H t}\right). \tag{3.22}$$

The differential decay rate of B meson (in a flavor eigenstate) that decays into the final state f can be obtained by evaluating Equation (3.12), where N_f is a normalization:

$$\begin{aligned} \frac{d\Gamma[B \to f](t)}{dt} &= N_f \left|\langle f \,|\, \mathcal{H} \,|\, B(t)\rangle\right|^2 && (3.23)\\ &= N_f \left| g_+(t)\left\langle f|B\right\rangle + \frac{q}{p} g_-(t)\left\langle f|\bar{B}\right\rangle\right|^2 . && (3.24) \end{aligned}$$

Of primary interest are decay rate asymmetries

$$\mathcal{A}_{\text{mix}}(t) = \frac{d\Gamma[B \to f](t) - d\Gamma[B \to \bar{f}](t)}{d\Gamma[B \to f](t) + d\Gamma[B \to \bar{f}](t)} \tag{3.25}$$

where some factors vanish and some systematic experimental effects cancel out.

When investigating the decay rate in a final state f, it is useful to define the decay amplitude

$$\begin{aligned} A_f &= \left\langle f|B\right\rangle, && (3.26)\\ \bar{A}_f &= \left\langle f|\bar{B}\right\rangle, && (3.27) \end{aligned}$$

which can then factorize out of decay rate asymmetries. With the simplification, that only flavor specific final states f_{fs} are allowed (which is useful for self tagging decays, used in the analysis of this thesis), the decay rates with usage of Equation (3.22) can be expressed as

$$\begin{aligned} \frac{d\Gamma[B \to f_{fs}](t)}{dt} &= \frac{1}{2} N_f |A_f|^2 e^{-\Gamma t}\left(\cosh\!\left(\frac{\Delta\Gamma}{2}t\right) + \cos(\Delta m t)\right) && (3.28)\\ \frac{d\Gamma[\overline{B} \to f_{fs}](t)}{dt} &= \frac{1}{2} N_f |A_f|^2 \left|\frac{p}{q}\right|^2 e^{-\Gamma t}\left(\cosh\!\left(\frac{\Delta\Gamma}{2}t\right) - \cos(\Delta m t)\right). && (3.29) \end{aligned}$$

Neglecting the decay rate difference, which is justified for neutral B mesons, since [27]

$$\Delta\Gamma_d/\Gamma_d = +0.001 \pm 0.010, \tag{3.30}$$

the probabilities $\mathcal{P}$ for a neutral B meson state to oscillate to the $\overline{B}$ meson flavor state or have the initial B meson flavor at time t can be obtained with

$$\begin{aligned} \mathcal{P}_{\text{osz}}(t) &= \left|\left\langle \overline{B}\middle|B(t)\right\rangle\right|^2 = |g_+(t)|^2 && (3.31)\\ &\approx \Gamma e^{-\Gamma t}(1 + \cos(\Delta m t)) && (3.32)\\ \mathcal{P}_{\text{n/osz}}(t) &= |\langle B|B(t)\rangle|^2 = |\frac{q}{p} g_-(t)|^2 && (3.33)\\ &\approx \Gamma\left|\frac{q}{p}\right|^2 e^{-\Gamma t}(1 - \cos(\Delta m t)) && (3.34) \end{aligned}$$

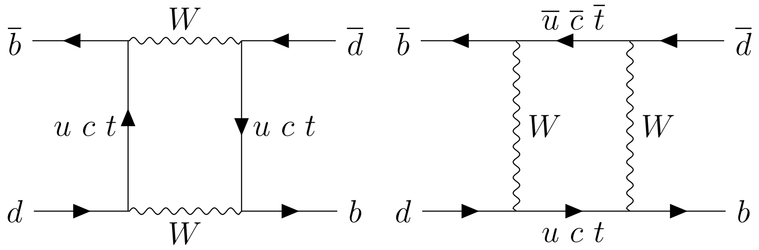

Figure 3.3.: Box diagrams for B_d^0 meson mixing. Mixing can occur via u, c or t loops, while the t loop is dominant. For B_s^0 meson mixing, the d quark can be replaced by an s quark.

3.2.3. Charge Parity Violation

There are many observations that suggest that processes happen with a different probability when transforming the constituents under the CP operations. While CP violation in the neutral Kaon and neutral B meson system has been firmly established, it was observed at LHCb in 2019 for the first time in charm decays [34]. B meson decays provide an optimal ground for precision measurements and are very well suited for the clean environment at lepton colliders.

In the SM, CP violation can occur in quantities with phase difference. Phases that are related to the CKM matrix are called *weak phases* ϕ_i. Additionally, there are phases which originate from hadronic interactions of rescattering processes of intermediate states. The corresponding parameters are usually referred to as the *strong phases* δ_i, since strong interactions play a dominant role.

A useful way in the flavor sector is to distinguish between three distinct occurrences of CP violation,

- CP violation in decay,
- CP violation in mixing, and
- CP violation in interference between mixing and decay.

For the *CP violation in decay*, the decay rate from a particle X differs from the charge conjugate reaction

$$\left|\frac{\Gamma[X \to f]}{\Gamma[\bar{X} \to \bar{f}]}\right| \neq 1. \tag{3.35}$$

CP violation in mixing occurs, when flavor eigenstates are not the mass eigenstates

$$\left|\frac{q}{p}\right| \neq 1. \tag{3.36}$$

For a measurement the decay chains $B \to \overline{B} \to \bar{f}$ and $\overline{B} \to B \to \bar{f}$ are compared. It can be expressed in a non-vanishing phase ϕ in the imaginary part of

$$\mathrm{Im}(M_{12}\Gamma_{12}^*) = |M_{12}||\Gamma_{12}|\sin\phi \tag{3.37}$$

leading to an asymmetry

$$\mathcal{A}_{CP_{\mathrm{mix}}} \approx -\frac{\Delta\Gamma}{\Delta m}\tan\phi, \tag{3.38}$$

which is rather small for B mesons. CP violation in mixing in neutral B_s^0 or B_d^0 systems hasn't been observed yet[27].

CP in interference between mixing and decay can occur, when a final state is accessible from both B meson flavors and both processes, $B \to f$ and $B \to \overline{B} \to f$ interfere with each other. When the condition $\mathrm{Im}(\lambda_f) \neq 0$ for

$$\lambda_f = \frac{\bar{A}_f}{A_f}\frac{q}{p} \tag{3.39}$$

is fulfilled, the so-called mixing induced CP asymmetry is

$$\mathcal{A}_{\text{mix.ind}}(t) = \mathcal{S}_f \sin(\Delta m t) - \mathcal{C}_f \cos(\Delta m t) \tag{3.40}$$

with

$$\mathcal{C}_f = \frac{1 - |\lambda_f|^2}{1 + |\lambda_f|^2} \tag{3.41}$$

$$\mathcal{S}_f = \frac{2\mathrm{Im}(\lambda_f)}{1 + |\lambda_f|^2}. \tag{3.42}$$

These time dependent quantities give an excellent opportunity for the search of new physics or precision measurement of participating parameters. The Belle experiment has observed mixing induced CPV in 2002 in the neutral B meson system [35].

3.2.4. Coherent *B* Meson Mixing

Experimentally, at Belle and Belle II, not a single B meson, but the decay of a $\Upsilon(4S)$ resonance in two B mesons is produced. In case of neutral B mesons, they are entangled as determined by quantum mechanics. For the interpretation of measurements with the time difference of both B mesons, it is necessary to develop a framework of this entangled state. For a more detailed description of coherent B meson mixing, the reader is pointed to[17]. At Belle and Belle II, B mesons are produced via the $\Upsilon(4S)$ resonance, with $J^{PC} = 1^{--}$ in an entangled state [36]. The wave function for the common coherent final state

$$\left|B(t_1)\overline{B}(t_2)\right\rangle = \frac{1}{\sqrt{2}}\left(|B(t_1)\rangle\left|\overline{B}(t_2)\right\rangle - \left|\overline{B}(t_1)\right\rangle|B(t_2)\rangle\right) \tag{3.43}$$

is antisymmetric, with an orbital angular momentum L=1 (P-wave). The state evolves coherently until one B meson decays. The time evolution can be described with

$$\begin{aligned}\left|B(t_1)\overline{B}(t_2)\right\rangle = \frac{1}{\sqrt{2}}([g_+(t_1)g_+(t_2) - g_-(t_1)g_-(t_2)]\,(|B\rangle\left|\overline{B}\right\rangle - \left|\overline{B}\right\rangle|B\rangle) \\ + [g_+(t_1)g_-(t_2) - g_-(t_1)g_+(t_2)]\,(\frac{p}{q}|B\rangle|B\rangle - \frac{q}{p}\left|\overline{B}\right\rangle\left|\overline{B}\right\rangle))\end{aligned} \tag{3.44}$$

A common decay rate $\Gamma(f_1, f_2)$ for the final states f_1, and f_2 at decay at time t_1 and t_2, respectively, is given by

$$\Gamma(f_1, f_2) = \left|\langle f_1, f_2 \,|\, \mathcal{H} \,|\, B_1 B_2\rangle\right|^2. \tag{3.45}$$

It can be described in relation to the relative time difference $\Delta t = t_2 - t_1$ [37].

For instance, when a process with one B meson, that decays in a CP eigenstate, where $B \to f$ and $B \to \overline{B} \to f$ is possible, is investigated, the time dependent CP asymmetry is

$$\mathcal{A}(\Delta t) = \mathcal{S}_f \sin(\Delta m \Delta t) - \mathcal{C}_f \cos(\Delta m \Delta t). \tag{3.46}$$

Integrating and normalizing over the complete phase space of $\Upsilon(4S)$, the probability $\mathcal{P}$ of obtaining a B or $\overline{B}$ meson with a charge of $q = 1$ or $q = -1$ respectively, is obtained by

$$\mathcal{P}(\Delta t, q) = \frac{1}{4\tau} e^{-|\Delta t|/\tau}(1 + q(\mathcal{S}\text{sin }(\Delta m \Delta t) - \mathcal{C}\text{cos}(\Delta m \Delta t)), \tag{3.47}$$

where τ is the lifetime of a B meson.

3.3. Flavor Tagging

In order to experimentally measure these asymmetries, the flavor of both entangled B mesons have to be known at their decay. While exclusive tagging methods, e.g. the Full Event Interpretation provide an excellent purity, the efficiency is in the lower percentage range for semi-leptonic and in the per mille range for hadronic decays.

In order to obtain a higher significance for a measurement from expensively recorded data sets, an inclusive method, which is able to assign almost every event an estimate of a probability for a certain flavor is favorable. This method is called flavor tagging and described in the following section.

3.3.1. Principles of Flavor Tagging

At Belle and Belle II, most of the recorded data is at the $\Upsilon(4S)$ resonance. With a fraction grater than 0.96 [27], the resonance decays into two B meson pairs. With a relative fraction of about $51.4 \pm 0.6\%$ over $48.6 \pm 0.6\%$, the decay into charged B mesons is slightly favored.

Usually, for an analysis, one or both B mesons are reconstructed. While a full reconstruction of the complete event is only possible in a small fraction of cases, it is often more feasible to reconstruct only one of the two mesons and tag the accompanying partner. Except for a experimentally possible cross-feed between both channels, in principle both reconstructed mesons can be treated independently.

In time dependent CP violation analyses (see Section 3.2.3), the time difference between the decays of both neutral B mesons is required. By measuring the distances between the decay vertices of both mesons and by knowing the boost $\beta\gamma$, the time difference

$$\Delta t = \frac{\Delta z}{\beta\gamma c} \tag{3.48}$$

can be reconstructed. In Figure 3.5 such a decay is displayed schematically.

For charged $B^{\pm}$ mesons, the determination of the flavor of the accompanying meson is quite simple, it can be concluded directly from the signal side flavor. Neutral B mesons present a greater challenge.

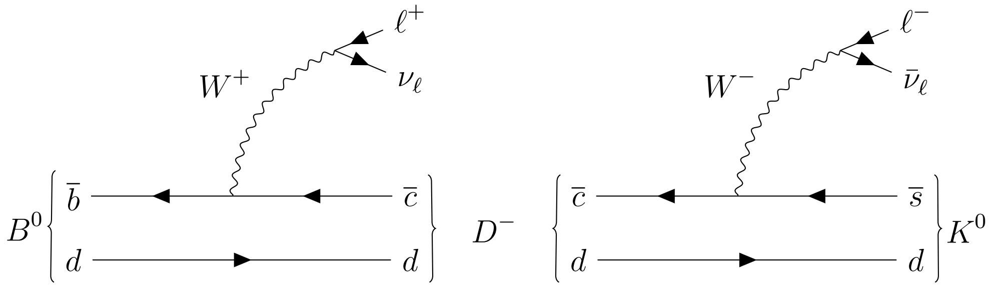

Figure 3.4.: Feynman Graph of a semi-leptonic B^0 meson decay. Here the the charm component X_c is an D^- meson (left) and a possible subsequent semileptonic decay. The transition of the bottom quark into a charm quark is mediated via weak interaction. The CKM matrix element, which governs this process is V_{cb}. The d quark is just a spectator quark.

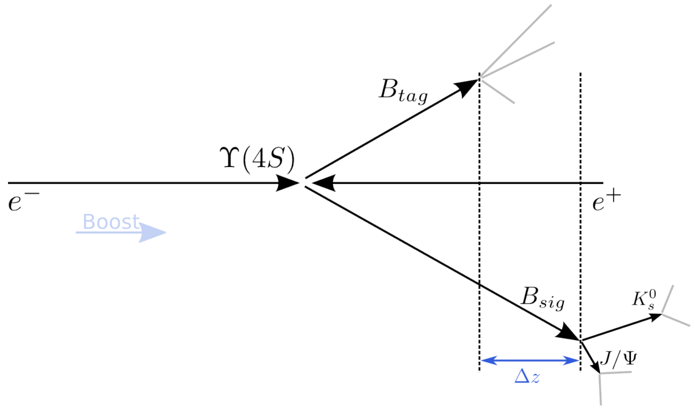

Figure 3.5.: Scheme of a typical particle decay event at Belle or Belle II. Here the $\Upsilon(4S)$ decays into two B mesons. The determination of the distance between the decay vertices for a typical $\Upsilon(4S)$ event at an asymmetric lepton collider allows to calculated the relative time difference Δt between both B meson decays. Taken from [38].

While the signal side B meson is reconstructed successfully, a probability of the flavor for the tag side B meson (B_{tag}) is estimated with a multivariate analysis method (MVA), more detailed in Chapter 4. For the different decay chains, the decay topology can be restructured in categories with different attributes, where the substructure show flavor specific characteristics. Most of them exploit correlations between charge, momentum and other attributes of one or more particles of the tag side in one event.

The presumably most powerful discrimination attribute are leptons from semileptonic decays

$$B^{\mathit{0}} \to X_c^- \ell^+ \nu_\ell, \tag{3.49}$$

where B mesons decay in a component with a charm quark X_c. Since the bottom quark decays via the weak interaction, the charge of the lepton is directly linked to the flavor of the bottom quark in the decay chain. The highest background for this category is caused by a successive decay of the charmed component. It can also decay semileptonically in a $c \to s$ transition,

$$X_c^- \to X_s^0 \ell^- \bar{\nu}_\ell \tag{3.50}$$

with the effect, that now the charge of the lepton indicates the opposite flavor of the B meson. An example for both processes is shown in Figure 3.4. The discrimination, or even the exploitation of both processes at once, can be done kinematically. Momentum, angular parameters and the determination of missing quantities, e.g. missing momentum

$$p_{\text{miss}} = p_B - p_X - p_\ell, \tag{3.51}$$

or missing angular momentum are correlated to the decay type. Statistically, primary leptons carry a higher momentum than secondary leptons.

Similarly, the kinematics are useful to classify *slow pions*. In hadronic decays, a high fraction of B mesons decays via $D^{*\pm}$ resonances to D mesons and charged pions. These pions are produced almost at rest in the $D^{*\pm}$ center of mass system, with a distinct direction to the D meson decay products.

Another very clean channel is the *kaon* category. The main source of kaon are $b \to c \to s$ transitions. The charge of a kaon produced in such a process directly indicate the flavor of the B meson. Furthermore the number of neutral kaons in an process can be a hint for the decay process. When observing one or more neutral kaons in one event, the probability is high that the observed kaons were produced by $s\bar{s}$ quark pair popping. There is also the possibility to obtain ‘wrong signed’ kaons, which were produced via of $b \to cW^-$ decays rather than directly from the $b \to c \to s$ chain.

Slow pions and charged kaons can occur in the same decay chain. Together with the angular momentum, correlations between both candidates can be used. Furthermore, correlations between fast and slow particles can be used opportunistically.

A fourth category are Λ baryons produced in $b \to c \to s$ decay chain. They leave a

$$\Lambda \to p\pi \tag{3.52}$$

signature in the detector. Although the decay rate is very low compared to other channels, the signature is very distinct.

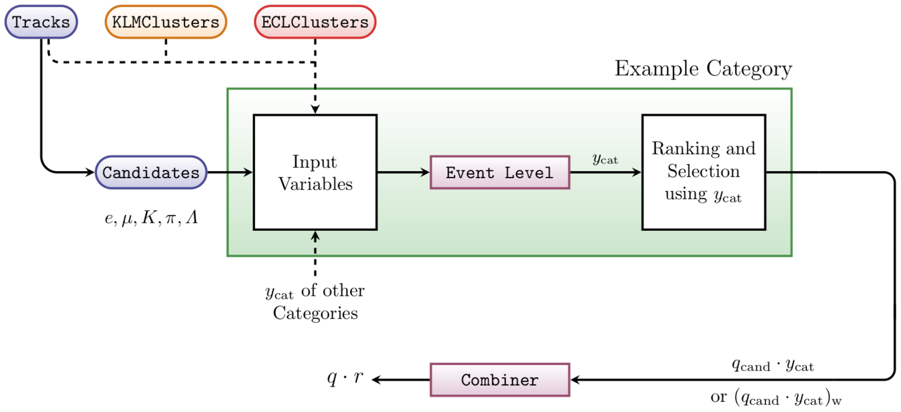
Tracks
KLMClusters
ECLClusters
Example Category
Candidates
e, μ, K, π, Λ
Input Variables
Event Level
y_{cat}
Ranking and Selection using y_{cat}
y_{cat} of other Categories
$q \cdot r$
Combiner
$q_{\text{cand}} \cdot y_{\text{cat}}$
or $(q_{\text{cand}} \cdot y_{\text{cat}})_{\text{w}}$

Figure 3.6.: Scheme for the category-based flavor tagger of Belle II. Taken from [41]

3.3.2. Expectations of Flavor Tagging

The development of B meson Flavor Tagging has been part of the experimental approach since the early stages of the Belle experiment, where originally a multi-staged approach was chosen [39]. The classification of one event is subdivided into several individual steps. In each event, particle candidates are treated under different hypotheses and assigned to different categories, which mimic several flavor specific aspects, described in Section 3.3.1. This can be for example the primary lepton category, which relies mostly on kinematic variables.

At the early stages of the algorithm, these assignments were determined purely by Monte Carlo based lookup tables. Later on, experiments with the usage of multivariate methods have been performed, which have the advantage being able to take correlations between features into account. The final version of the Belle flavor tagger achieved an effective tagging efficiency on data of

$$Q = (30.1 \pm 0.4)\% \tag{3.53}$$

for roughly 1.5 million events in multiple semi-leptonic and hadronic control channels[2, p. 108]. For the Belle II category based flavor tagger, no reference value of effective tagging efficiency on data has been published, at the time of writing.

The current category-based tagger for the Belle II experiment approach [40, 41] uses gradient boosted decision trees for classification of the sub-categories. The respective subcategory does not necessarily obtain the B meson flavor as the target value for the classifier optimization. Here, the affiliation of a particle candidate to a certain sub category is evaluated. The algorithm also takes associated KLM clusters and ECL clusters into account. For each candidate, the most relevant categories are determined and the results are combined with another gradient boosted tree with the truth value of the B meson flavor as binary target. This approach is schematically shown in Figure 3.6.

In the next chapter, a novel approach for flavor tagging is shown, which is based on a deep neural network.

4. The DNN Flavor Tagging Algorithm

The process of flavor tagging, the assignment of a flavor to B meson, is a complex task. As shown in the previous section, there are vast possibilities of different decay chains, with unique signatures and attributes. These different decay topologies can be exploited for the flavor estimation. Unlike the category based approach shown in the previous section, no handcrafted features are used extensively. Instead, an approach is shown which relies on low level features to construct a representation of the provided data.

It will be discussed in the following chapter, how a multilayer perceptron can successfully be used to improve the state of the art tagging algorithm while relying on low level features only. The aspects of the algorithm composition and training procedure are discussed. A good resource for machine learning in general is [42, 43], in which the definitions of the next section can be found.

4.1. Introduction to Machine Learning

The description of nature - the world around us, in a meaningful way, is a difficult task. Models are required for the description and prediction of processes. Observables, quantities that can be measured within a certain accuracy, allow us the parametrization of specific states. More complex objects, which unify multiple aspects of a certain state, can be constructed out of multiple parameters. For instance, a book can be described by its constituents, an object with pages and written letters on it – it could just be described by its shape, or even more abstractly, by its functionality. The way how an object is described by its parameters is called the representation of an object. There are representations which are more or less suited for these object descriptions and it remains key to find a good representation.

Machine learning is the process of an algorithm of learning how to solve a specific task without being explicitly programmed how to do so. These algorithms are data-driven and there are many techniques to depending on the class of problems and available data sets. Finding a good representation of the data is one of the most important aspects of solving those tasks. Usually, multivariate techniques are used, which exploit statistical attributes, for instance correlations between the observed features. The process of parameter adaptation of the model on data is also referred to as training of the algorithm.

In many cases it is initially unclear, if and to what extent the features are correlated. Still the probability distribution of the features can be statistically approximated. In the best case these are optimally used by the algorithms to fulfil their tasks.

In a statistical analysis, it is tried to determine and exploit those correlations to find a model of the underlying probability distribution.

Machine learning has been applied successfully in a vast range of areas. For instance, these models can be used prediction of protein structures [44], predictions of long-term structures of glasses [45] or candidate selection in reconstruction of particle decays.

Two important machine learning applications are classification and regression. In classification, a set of input parameters is mapped to a set of classes and the probability distribution for each class is estimated. In regression, a continuous function maps one or many input parameters to a distinct set of output parameters.

The three most widespread types of machine learning categories are supervised, unsupervised and reinforcement techniques.

- *Unsupervised techniques* do not use labeled data, meaning there is no “truth” information available during the training process. They are used to measure the distance between classes, to perform clustering analysis or outlier detection. One example of an unsupervised algorithm will be used in Section 4.4.
- *Supervised techniques* use labeled training data. Here, the solution for a specific task is available during the training process and can be used to improve the results. Regression and classification fall under this category. For some problems, for instance a particle decay and its detector interaction, training data can be generated using Markov Chain Monte Carlo methods. Simulated data relies on the accuracy of the model on which the data was generated. Since those models do not necessarily mimic nature with the required precision, a calibration and domain adaption methods are favorable.
- In *reinforcement learning* an autonomous agent interacts with an environment to maximize its reward. It allows directions in the action space of the algorithm, where the reward of the next position is lower, although it maximizes the final reward. For instance, a policy to evaluate the quality of a state can be introduced and optimized.

A major issue all these techniques have to cope with when operating in a high dimensional feature space is the so-called *curse of dimensionality.* A data point with d features can be pictured as a vector in a d dimensional space. Data points with similar features end up in a similar region in the feature space and changing the configuration of those features will push the location to a different place. The number of possible configurations does not increase linearly but exponentially with increasing the dimension d. Therefore, the relative distance between data points within similar classes is similar to the distances between data points of different classes when there are a large number of features. Also, with increasing d, a increasing number of configurations have to be shown to the algorithm during the training to allow it to interpolate between known regions in the feature space. This effect is enhanced when unnecessary features with low or negligible correlations are used to enrich the feature space.

The choice of suitable *model complexity* involves trade-offs described by the *bias-variance dilemma* [43]. For instance, the truth y can be decomposed into two functions, as $y = f(\mathbf{x}) + \epsilon$. Here, the function f depends on the features $\mathbf{x}$, while ϵ does not and therefore is not predictable by an algorithm from $\mathbf{x}$. The goal of the algorithm is to approximate the function f. The function estimator denoted as $\hat{f}$. The precision of the approximation, can be separated into two major distinct parts – bias and variance. The bias of the estimator can be described as

$$\mathrm{bias}(\hat{f}) = \mathbb{E}\left[\hat{f} - f\right], \tag{4.1}$$

where $\mathbb{E}$ is the expectation value taken over a distribution of samples. The variance of the estimator is given by

$$\mathrm{var}(\hat{f}) = \mathbb{E}\left[(\hat{f} - \mathbb{E}[\hat{f}])^2\right]. \tag{4.2}$$

The mean squared error, which is given by the quadratic deviation of the approximation,

$$\mathbb{E}[(\hat{f} - f)^2] = \mathrm{bias}(\hat{f})^2 + \mathrm{var}(\hat{f}) \tag{4.3}$$

is a good metric for a comparison of the optimization process. Note that this does not include the irreducible part ϵ in the truth information.

If the model complexity is chosen to be too shallow, the bias is large. The model cannot be adapted to the underlying data structure in the required way and correlations might not be captured correctly. In contrast, if the model complexity is chosen to be too high, this might lead to *overtraining* during parameter adaptation. This leads to a large variance. The algorithm is now oversensitive to statistical fluctuations in the training data set and the model is not able to generalize predictions in the required way. Methods for finding the right trade-off between both errors are discussed in Section 4.2.

While techniques as shown above can be employed, in practice, a more specified architecture and a multiple stacking of processing layers in the algorithm is required. Each of these layers include linear or non-linear transformations or mappings. This class of computational models is referred to as *Deep Learning*[46]. The main goal is to find and improve a *representation* for the investigated data set, with increasing complexity in each layer of the algorithm. In a multilayer architecture, the current layer is able to re-use features, which are found by the previous one. There are multiple attributes, which describe ‘good’ representations [47]. One prerequisite to successfully apply deep learning algorithm is a sufficient amount of training data.

An architecture is chosen, that is able to encode a representation with the following properties:

- It is *invariant* to local changes in the object space. For instance, for the classification of an object, this can be the strength of a shadow of that object in the picture.

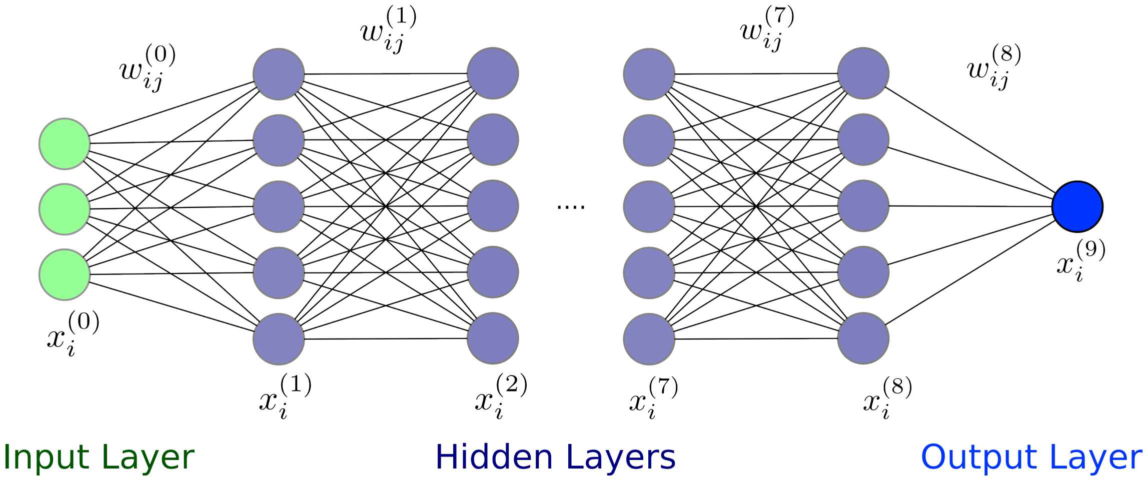

Figure 4.1.: Schematic view of a multilayer perceptron. Taken from [38]

- Unique features of this representation should be *disentangled*, the classification should not fail just because there is one independent feature missing in the sample. In a particle physics decay, this could be an unclear or missing the particle identification of a less important particle.
- Furthermore, an *expressive* representation is favorable, which means, that the number of possible regions in the solution space should be maximized. For instance, Gradient Boosted Decision Trees with N nodes allow only $\mathcal{O}(N)$ regions. A dense representation can cover up to $\mathcal{O}(2^N)$ different regions.

A multilayer architecture is able to learn these abstract concepts. The approach to apply a such a technique for Flavor Tagging is presented in the following section. One major goal is to obtain a feature representation that super seeds hand-crafted features in time and classification efficiency.

4.2. Model Architecture

For the classification process, a multilayer perceptron architecture (MLP) is used. The first version of flavor tagging procedure and algorithm for the application of a deep multilayer perceptron was in introduced in [38, 48]. During this thesis the algorithm was completely rewritten in `Tensorflow` [49] with several minor adaptations. A schematic view for an MLP is provided in Figure 4.1.

A multilayer perceptron is composed of three different layer types – an input layer $x_i^{(0)}$, hidden layers and an output layer k. Each layer contains an adjustable amount of nodes $x_i^{(k)}$, the dimension of the input layer is determined by the dimension of the feature space. Each node has a bias attribute w_{i0} and is connected to the previous layer via weight parameters w_{ij}. These parameters are adjustable and are fitted during the training procedure on a dedicated training data set. Parameters regarding the network architecture and which are pre-defined during the fitting procedure are so-called *hyper-parameters*. The can be optimized separately in so-called *hyper-parameter searches*, where multiple techniques, such as *grid search* or *bayesian* approaches [50] can be used.

Each node is transformed by a non-linear activation function σ with

$$x_i^{(k+1)} = \sigma^{(k+1)}\left(\sum_{j=1} w_{ij}^{(k)} x_j^{(k)} + w_{i0}^{(k)}\right), \tag{4.4}$$

usually the sigmoid, hyperbolic tangent or rectified linear unit functions are used. Using a non-linear activation function allows the representation of non-linear relations.

During the training of a multivariate classifier parameters $\mathbf{w}$ of the model $\hat{f}$ are adapted on a training data set $\mathbf{x}$ with truth labels $\mathbf{y}$ by minimizing a loss function $\mathcal{L}$.

For the comparison of the two probability distributions P and Q, which represent the true and estimated probability distributions, the cross-entropy $H(P,Q)$ with

$$H(P,Q) = -\mathbb{E}_P[\log Q] \tag{4.5}$$

is a common metric to quantify the differences.

The loss function for a binary classification task with $y_n \in \{0,1\}$ can be written in terms of the binary cross-entropy

$$\mathcal{L}(\mathbf{w}) = -\sum_{n=1}^{N}[y_n \log \hat{f}_n(\mathbf{w}) + (1-y_n)\log\Big(1-\hat{f}_n(\mathbf{w})\Big)] \tag{4.6}$$

calculated for a sample size N. Here y_n and $\hat{f}_n$ represent truth values and estimated values for each data point respectively.

The loss function is minimized with an *gradient descent* algorithm on a *mini-batch*. The batch size itself is a hyper-parameter to adjust the robustness against statistical fluctuations on the training data set. The minimization procedure can be modified by using an additional momentum term which takes into account the weight update from the previous optimization step $\mathbf{w}_{\text{prev}}$. This speeds up the process, while moving along ravines in the multidimensional weight space. The parameters then are updated with

$$\Delta\mathbf{w} = -\eta\nabla\mathcal{L}(\mathbf{w}) + \mu\Delta\mathbf{w}_{\text{prev}}, \tag{4.7}$$

where η and μ are hyper-parameters, which are adjusted dynamically during the training procedure.

The gradients for the weights can be efficiently calculated by *backpropagation* [51] based on applying the chain rule

$$\frac{\partial\mathcal{L}}{\partial w_{ij}^{(k)}} = \frac{\partial\mathcal{L}}{\partial a_j^{(k)}}\frac{\partial a_j^{(k)}}{\partial w_{ij}^{(k)}} \tag{4.8}$$

$$a_i^{(k)} = \sum_j w_{ij}^{(k)} x_j^{(k)} \tag{4.9}$$

and reduces the computational complexity to linear order of the network parameters.

For prevention of over-adaptation during the training process, regularization mechanisms are required. Regularization mechanisms allow to deal with the bias-variance dilemma in the following way: The number of model parameters is chosen to be large, such that the variance dominates the error budget. Regularization the moves the error of the model towards a larger bias until the bias becomes comparable with the variance. Two mechanisms are used for regularization, weight decay and early stopping.

The loss function $\mathcal{L}$ can be expanded by a regularization term

$$\mathcal{L}_{\text{WD}} = \mathcal{L} + \frac{\alpha}{2}||\mathbf{w}||^2. \tag{4.10}$$

Here, $||\mathbf{w}||$ corresponds to the norm of the weight tensor. High weights during the training process are penalized. This procedure is referred to as the so-called L^2 *weight decay*.

For another regularization mechanism, the loss function is monitored during the training procedure on a separate data set. Usually, before the training process, the available data is split into a training, validation and test set, to ensure that a stable solution has been found during the process. Once the loss function stops decreasing on the validation set and starts to increase on the training data set, the training procedure is halted. *Early stopping* is a efficient way to prevent and monitor overfitting during the training procedure.

The loss function of this approach is not necessarily convex and a convergence of the algorithm is not necessarily guaranteed.

4.3. Feature Space

The selection of an optimal feature set is crucial for the performance of the classifier. On one hand, neglecting important features might lead to the loss of important correlations, and therefore decrease the classifier performance. On the other hand, selecting unimportant features is unfavorable due to the curse of dimensionality, see Section 4.1.

For machine learning in particle physics, feature selection should be sensitive to the kinematic attributes of the Flavor Tagging procedure. The most important variables are *charge* and *momentum*. Almost every conclusion regarding the physical domain relies on those quantities, including the most prominent categories – primary and secondary particles, see Section 3.3.1. The momentum is transformed to spherical coordinates in the center-of-mass system (CMS) of the $\Upsilon(4S)$ resonance. For each particle, the combined likelihood ratios for *particle identification* of multiple possible particle hypothesis are used, including electrons, kaons, pions, protons and muons. To allow possibly an inference about common vertices, the *impact parameters* of each track are included. Since a decay is not explicitly reconstructed on the tag-side, no vertex fits can be provided. The *p-value of the track fit* and the available *number of hits* in the tracking detectors are included as a measure of the quality of the track fits. This summarizes to an input dimension of the feature space of $d = 120$, the charge is implicitly integrated by an external ordering of the input features - they are grouped by charge.

To exploit the most important aspects of the flavor signatures and provide a better interpretability of the input vector by the network, the input is grouped by charged and sorted by momentum. The scheme is shown in Figure 4.2.

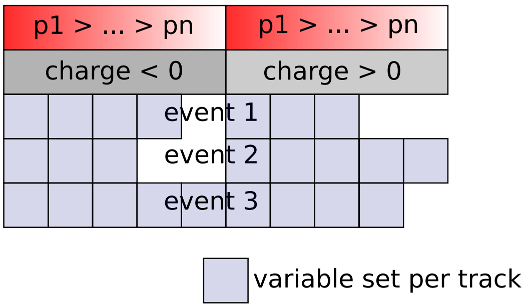

Figure 4.2.: Scheme of multiple input vectors of the Neural Network. Taken from [38].

The input is preprocessed in a way, that it operates naturally in a regime of the activation function. With a *equal frequency binning* transformation, it is uniformly distributed in $x_i^{(0)} \in [-1, 1]$. In many cases, an event has less then ten charged tracks. Subsequently, the missing entries for the specific tracks are set to zero, which is the mean of the transformed distribution.

4.4. Representation Learning

Visualizing this high dimensional representation of the feature space in a two dimensional mapping can provide useful information hint about the distribution of distinct classes in the feature space.

The *t-distributed stochastic neighbor embedding* (t-SNE) is a technique to show similarities and cluster objects with similar attributes [52].

The solution in a low dimensional space is obtained by minimizing the Kullback-Leibler divergence (KL) between the joint probability distribution in the high dimensional space P and the low dimensional space Q,

$$\mathrm{KL}(P||Q) = \sum_{i,j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{4.11}$$

where p_{ij} and q_{ij} are probability distribution in the high and low dimensional space.

The probabilities $p_{j|i}$ for a similarity in the high dimensional space for objects $\mathbf{x}_i$ with a gaussian distribution σ_i

$$p_{j|i} = \frac{\exp\left(-||\mathbf{x}_i - \mathbf{x_j}||^2/2\sigma_i^2\right)}{\sum_{k\neq i} \exp\left(-||\mathbf{x}_i - \mathbf{x}_k||^2/2\sigma_i^2\right)} \tag{4.12}$$

is symmetrized with

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}. \tag{4.13}$$

For the probabilities in the lower dimensional space $\mathbf{y}_i$, a Student t-distribution

$$q_{ji} = \frac{(1 + ||\mathbf{y}_i - \mathbf{y}_j||^2)^{-1}}{\sum_{k\neq l}(1 + ||\mathbf{y}_k - \mathbf{y}_l||^2)^{-1}} \tag{4.14}$$

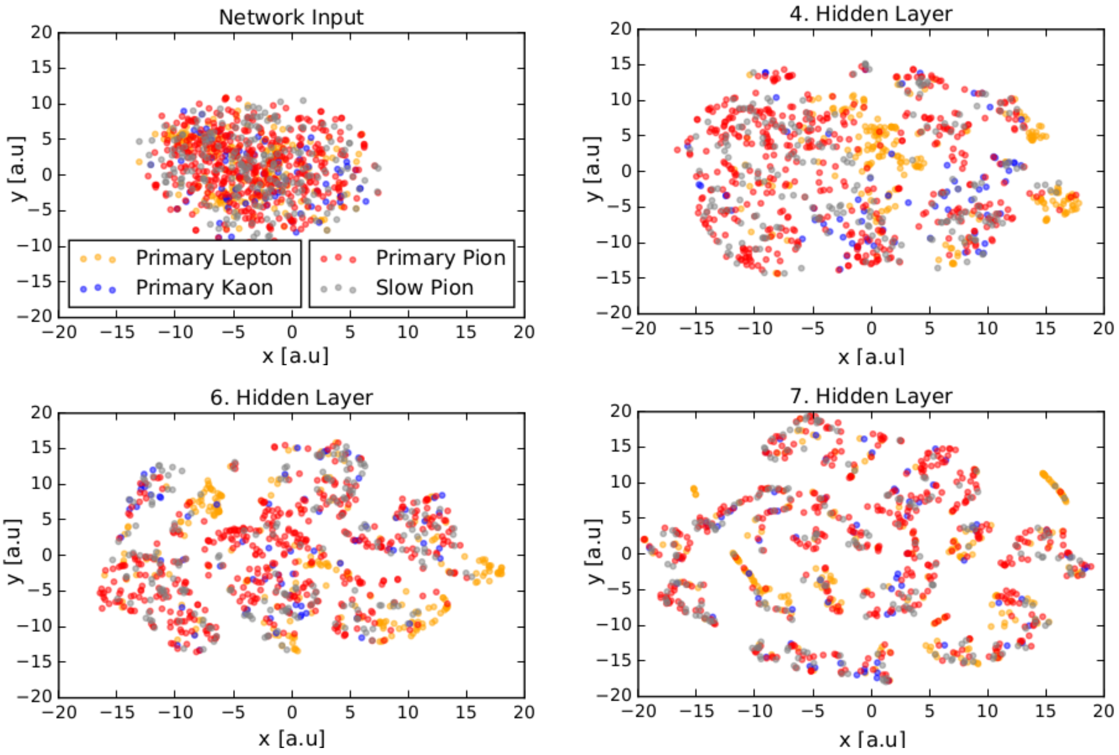

Figure 4.3.: T Stochastic Neighbor Embeddings for the T Stochastic Neighbor Embeddings for the input feature space, as well as the 4th, 6th and 7th layer of a fully trained MLP. Four distinct categories are colorized on truth level information. The clustering increases for each layer. The plots are in the $\mathbf{y}_i$ feature space with arbitrary units.

is used. Compared to a gaussian probability distribution this has the advantage, that possible clusters are less centered, sometimes described as the *crowding problem* taking advantage of the tailed distribution in the low dimensional space.

In Figure 4.3, the t-SNE distribution for the input and the output after different layers for a fully trained MLP for Flavor Tagging, with eight hidden layers is shown. For the preparation of the data sample considered in this section, four clearly separable distinct categories have been chosen – primary leptons, primary pions, primary kaons and slow pions.

While for the input space(Figure 4.3, upper left), the t-SNE algorithm fails to establish a clustering in the data set, for increasing depth of the neural network, a separation is clearly visible (e.g. Figure 4.3 lower right). Although t-SNE mappings cannot be seen as a proof for increasing class separability, this is still a strong indication, that the feature space of the DNN is evolving.

For the output of the final hidden layer, shown in Figure 4.4, a clear separation between the clustering is visible. Changing the overlay to the target categories, as shown at the bottom graphic of Figure 4.4, a clear majority of each cluster belongs to a certain B meson flavor. On the top graphic of Figure 4.4, two clusters with a majority of primary leptons are visible, which belong to different flavors. The primary lepton category is also supposedly the most powerful separation category for the category based flavor tagger.

To finalize, although not conclusive, the observations on this sample are in agreement with the assumption, that the algorithm is able to develop a feature representation, which resembles flavor specific categories. These categories are not necessarily congruent with the categories of the established tagging algorithms. This indicates, that a combination of both algorithms could even improve the class separability of the task. This assumption is investigated in Chapter 6.

4.5. Training and Performance on Simulated Samples

The flavor tagging algorithm is trained on so-called *mono-generic* Monte Carlo (MC), where only one B meson decays in the generic decay chain, and the other side to detector-invisible particles, here two neutrinos. Studies have shown, that even when applying the algorithm on signal-specific decay channel, the performance of the algorithm does not increase [38]. For the training procedure, a total sample size of roughly 13 million events is used. The sample is split and the loss function monitored on an independent test set with approximately one million events. The training is performed for a minimum number of epochs. If the loss function increases or not decreases below a certain threshold for a fixed number of epochs on the validation set, the training is stopped. The number of epochs and final performance is slightly dependent on the seeding of the initial values of weights and biases, and the constitution of the samples. The classifier state with the best performance is chosen for inference. In Figure 4.5, the loss functions on the validation and training data set are shown for different seedings of initial values. The absolute values of the training loss is below the validation set, which indicates a slight but expected over-training on the training data-set. The extent of overtraining is acceptable, because further

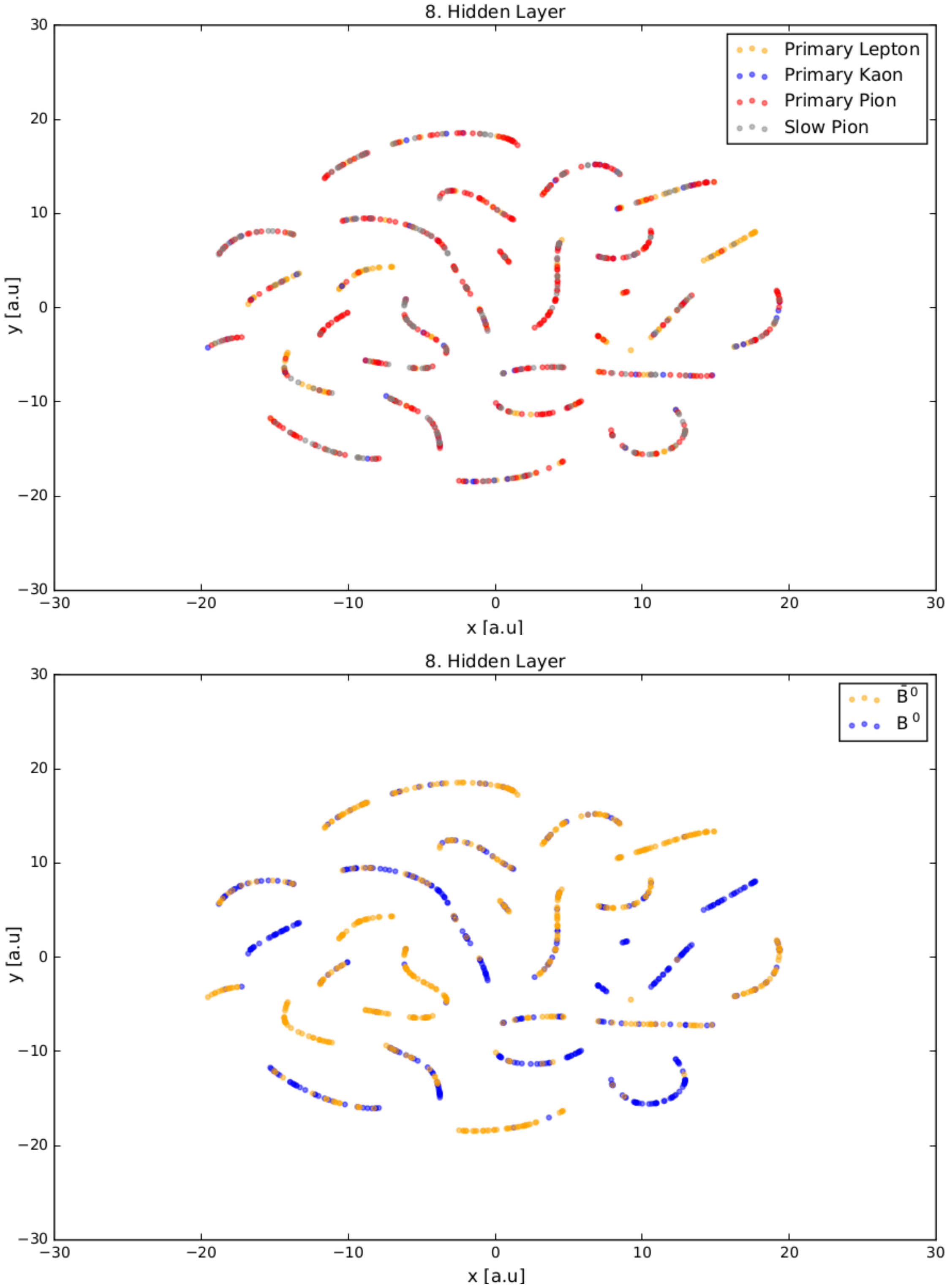

Figure 4.4.: T Stochastic Neighbor Embeddings for the feature space after the last hidden layer for a fully trained MLP. For the target category - the classification into a B and $\overline{B}$ meson - there is a clear clustering visible. The plots are in the $\mathbf{y}_i$ feature space with arbitrary units.

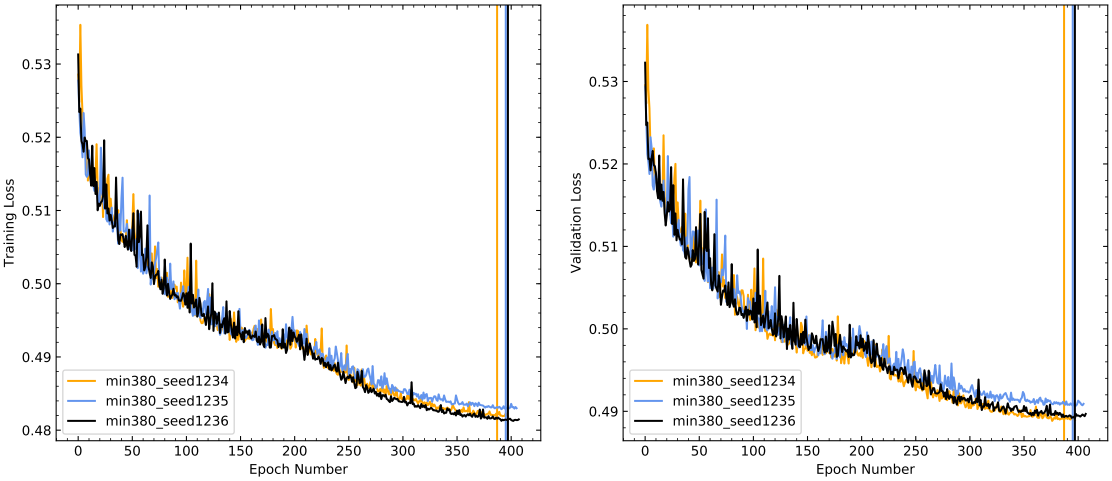

Figure 4.5.: Loss values on the validation and training data set for multiple seeds on the same data set. The best epoch is marked by a vertical line. The differences between the trainings originate from a different seeding of the initial values.

parameter adaptation still leads to an increase of the metric on the independent validation set. The epochs with the lowest classifier loss values on the validation set are marked. Subsequently, the training is stopped shortly after reaching these values.

The classifier shape of a fully trained classifier is shown in Figure 4.6 on the right hand side, both B meson flavors are shown separately. For visualization and validation an additionally generated Monte Carlo data set is used. In general, in spite of regularization, there might be slight overtraining effects for the MVC. These have to be identified and corrected. Therefore, the signal purity

$$p = \frac{N_B}{N_B + N_{\overline{B}}} \tag{4.15}$$

is evaluated. For a idealistic perfectly trained classifier, the purity should correspond to the probability for obtaining the exact fraction of signal events in a certain bin. In the upper half of Figure 4.6 on the right hand side, it is shown that this is almost the case. Deviations can be taken into account and a transformation be defined to correct for the discrepancies. Coefficients for the transformation function are determined on an additional and sufficiently large Monte Carlo data set, in order to be resistant against statistical fluctuations and to reduce the over-training effects.

Generally, further optimization of the algorithm could provide a boost in training time and accuracy of the algorithm. Adding additional features could increase sensitivity to additional physical effects and characteristics of the decay topology.

4.6. Alternative Approaches

The DNN approach can be compared with the category based algorithm, which was described in Section 3.3.1. The shape and the purity of the MC, with the official weight files from `BASF2` release `rel-01-02-04` are shown in the lower half of

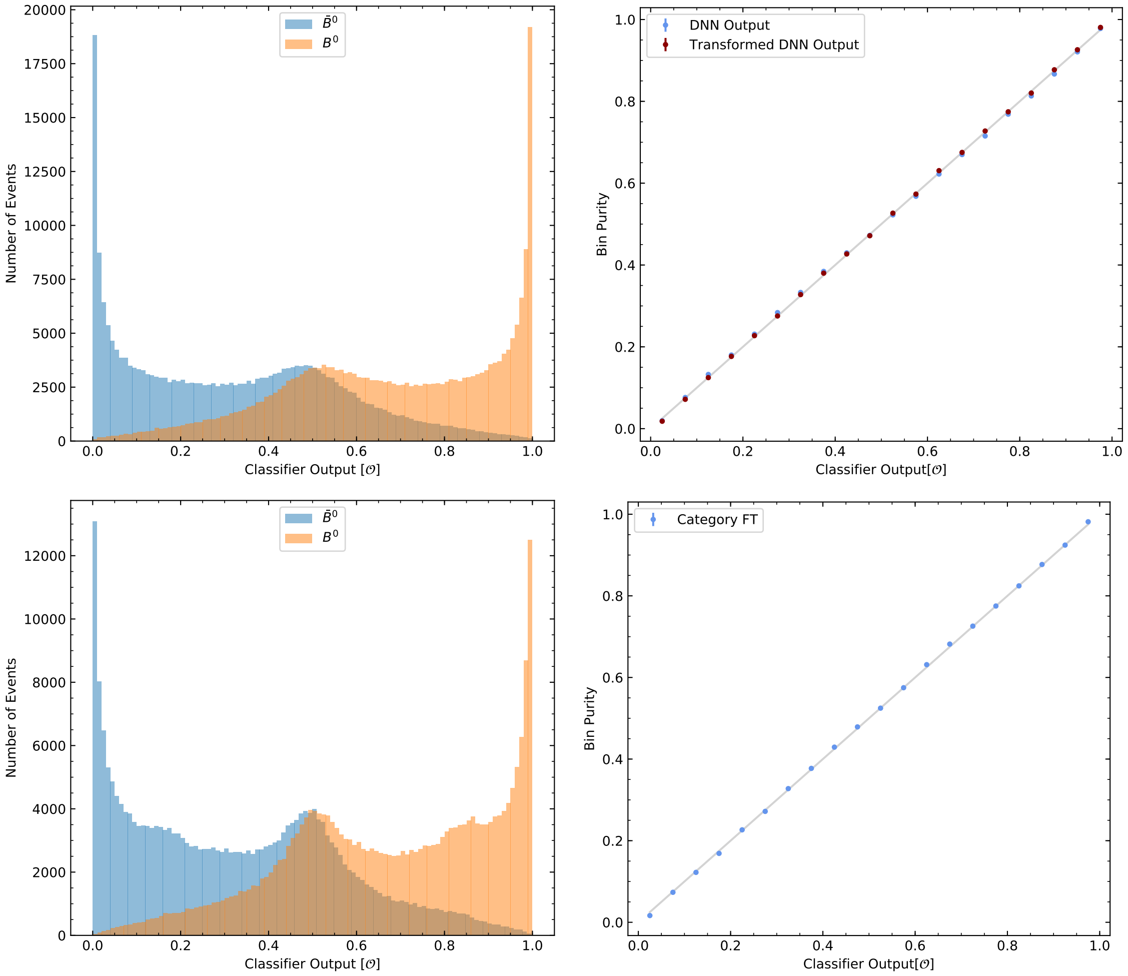

Figure 4.6.: Classifier shapes on a mono-generic test set with 500.000 events (left hand side), for B and $\overline{B}$ mesons separately. The sample is separately produced from the training and validation set. The purity for the classifier is shown on the right hand side, before and after an optional transformation, which readjusts for overtraining on a separate Monte Carlo data set. The deep neural network approach is shown in the upper half, the category based approach is presented in the lower half.

Classifier Name	Channel	Effective Tagging Efficiency $\mathcal{Q}$
Deep Neural Network Flavor Tagger	$B^0 \to \nu\bar{\nu}$	0.3455 ± 0.0024
Category Based Flavor Tagger	$B^0 \to \nu\bar{\nu}$	0.3419 ± 0.0024

Table 4.1.: Performance for the DNN and category based flavor tagging algorithm on mono-generic Monte Carlo data. A definition of the metric, the effective tagging efficiency $\mathcal{Q}$ is described in detail in Section 5.4. The DNN algorithm here shows a better performance.

Figure 4.6 for the same mono-generic test data set. The classifier has a different efficiency in specific ranges of the output. The selectivity for certain categories have a visible impact on the shape of the classifier distribution. The influence of a distinct kaon category is clearly recognizable in the range of output at 0.1 and 0.9 respectively with an increased enrichment of these events. A comparison of the performance of both flavor tagging algorithms is shown in Table 4.1. To determine the actual performance on data, this effective tagging efficiency has to be calibrated, which is described in the following chapter.

5. Calibration of the DNN Tagging Algorithm on Belle Data

Monte Carlo Studies are essential for the development process of the flavor tagging algorithm and of vital importance to get a first impression for its performance. Monte Carlo simulations are based on models, which are an inherently imperfect representation of nature with known limitations. The need to quantify potential differences between data and Monte Carlo is a central point in the reasonable usage of multivariate classifiers in a high precision field and is required for the interpretation of future measurements using those tools. Since multivariate methods are able to exploit higher order correlations between features, a validation and calibration on data is crucial to quantify how discrepancies have an impact on the predicted result. The systematic effects have to be investigated and understood.

Due to the great variety of possible decay chains, shown in Section 3.3.1, the tagging process must be validated and checked in a statistical and automated analysis.

The effective tagging efficiency $\mathcal{Q}$, a metric which enables the comparison of the performance of flavor tagging with respect to the uncertainty of future measurements, is defined in Section 5.4.1. For the calibration, the wrong tag fraction and the effective tagging efficiency are evaluated in four independent channels, and compared with the category based approach. For the measurement of the neutral B meson channels, self-tagging decays are used, where the signal side decay is flavor specific.

Of course, on data the truth information of the tag side is not known, but has to be inferred from the signal side. For neutral B mesons, the time integrated fraction of mixed events χ_d has to be taken into account as prior knowledge to reconstruct the fraction of the flavor on the signal side.

For the calibration on data, a multitude of tests and studies have to be performed. The following chapter will show why such a careful calibration is mandatory and show how a Deep Flavor Tagging algorithm performs on the data domain.

5.1. Analysis Framework

The calibration of the Deep Flavor Tagging algorithm is one of the first complex analyses on converted Belle data (see Section 2.3.1). The Belle II analysis framework (`BASF2`) did not provide many rudimentary functions at the start of this analysis, therefore it was necessary to add the missing functionalities and abstract certain analysis processes. Such a framework is developed in this thesis and is presented in the following sections.

The concepts of this framework and `BASF2` are fundamentally different. While `BASF2` is designed to implement and execute linear python scripts — the so-called steering files — for a more complex analysis, a deeper level of abstraction is required. Therefore, a separate validation framework was developed that also provides the possibility to create and execute `BASF2` processes in a more managed and controlled way. Several analysis steps are streamlined more efficiently. Nevertheless some extensions to `BASF2` were necessary and implemented.

The range of tasks of the framework includes administration, processing and monitoring of each production and preparation step. Consequently, this requires a high degree of automation of the processes. As a result of the automation a complete overview of the procedure is available. Furthermore it allows a transparent reproducibility of the analysis.

The framework is able to deal with an unreliable computing grid; each task is individually checked for errors and re-run if required. Multiple samples with slightly different configurations can be produced quickly and the risk of improper execution of scripts is minimized. For instance, samples which are required for new training and validation sets on different versions of the processing software, which would be required for the training of the Flavor Tagger on different `BASF2` versions. The tool is completely `Python` based and can chain multiple computing sites with different linux distributions together. The configuration is centralized and can be quickly adapted.

The analysis procedure is subdivided into several independent tasks. Each task has a specific configuration and may have an arbitrary amount of parent data sets. Each signature of the parent data set is generated completely from the used source files and the individual configuration. In this way, each task can be mapped to a unique identifier, which can be used to define the output files of a task. The processing is chained in a directed acyclic graph.

The unique identifier can be a hash function, which is calculated from the task type, the task parameters and the identifiers of the parent data set. Changing only one parameter results in a different unique identifier, and the complete chain will have to be processed again. For example as a simple case, one task could be the preparation and execution of several batch scripts, setting up the specific environment variables, and chaining and linking the required files automatically.

The processing of a task is subdivided into several jobs, which are sent to the computing grid and monitored automatically. Once all jobs are finished, an automated method is applied for checking the sanity of the task. This includes parsing the log files for errors, checking file sizes and certain file attributes. If an error is discovered, failed jobs are automatically resubmitted and supervised again. The complete operation is *atomic.* This means that the status of the task and all produced files will only be updated, once the processing is completed successfully.

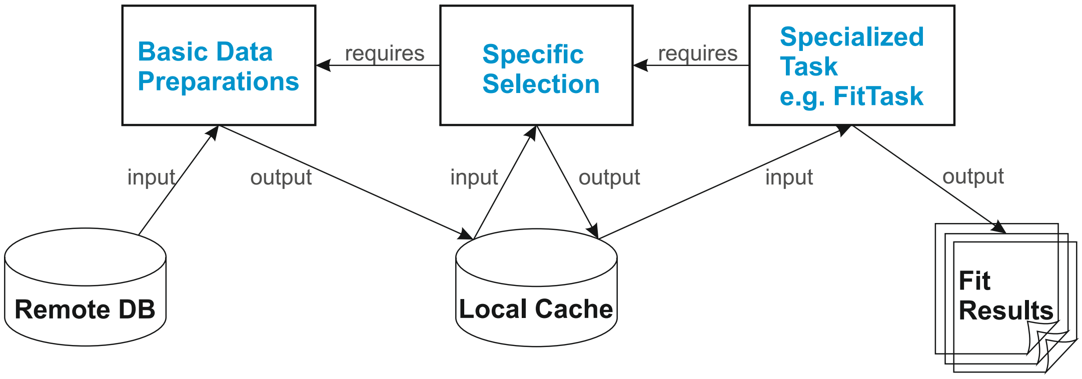

Figure 5.1.: Scheme of a simplified workflow scheduled with `luigi`. Tasks (in blue) access data sets to generate new outputs. The tasks have information about their dependencies. In this example, a task accesses a remote data store, applies some changes and stores these changes locally. Then a more specialized task, e.g. a variable selection task, prepares the data set for a fitting task.

Several different tasks are chained together with the pipelining framework `luigi` [53]. An example for the work flow is provided in Figure 5.1. The `Python` module provides the skeleton for creating specific task objects, which are specified by its own parameter set. At runtime, a scheduler checks if there are already successfully completed tasks, or if some still have to be processed. Usually the tasks verify if certain output files are available, to set their state to completed. Besides that, `luigi` can also handle the resource management, take care of the scheduling and provide a decent visualization of the workflow and dependencies. This has multiple advantages:

- If one or multiple parameters of a task have been changed, the specific tasks are run again and the complete chain is updated.
- A parameter change in a parent task does not necessarily mean that all child tasks but only a selection has to be executed again. This can save a lot of time and resources.
- It allows pre-caching files of computing intensive tasks for rapid development.
- Each analysis step is completely transparent and reproducible.

The configuration for all tasks and analysis steps and functions is separated from the execution code. This allows easier maintainability, adaptability and it is less prone to error. Every step of an analysis is easy to follow and can be investigated. Every data set is linked to a specific task and parameter configuration.

The analysis can be subdivided into two steps, an *online* part and an *offline* part. Configurations for online and offline analysis steps are handled separately. While the online part is mostly executed at the KEK computing center in Japan (data locality), the final steps of this analysis are performed at the local computing cluster (offline analysis). The online analysis part contains Monte Carlo generation, detector simulation, reconstruction and data pre-processing. Also all processing steps, which are performed with `BASF2`, are executed at KEK. During this step, a specific set of pre-cuts is applied, which reduces data sets to a manageable size. All channels are reconstructed in one processing step to reduce resource consumption. Data files have to be requested from tape storage only once.

Table 5.1.: Monte Carlo streams and samples provided by the Belle collaboration

signal-type	event-type	number of samples	weight
$B^+ \to X_c$	evtgen-charged	10	1.028
$B^0 \to X_c$	evtgen-mixed	10	0.972
$e^+ \ e^- \to c \ \bar{c}$	evtgen-charm	6	1.0
$e^+ \ e^- \to q \ \bar{q}$	evtgen-uds	6	1.0
rare B decays	special-mixedrare	1	0.02
rare B decays	special-chargedrare	1	0.02
$B \to X_u \ell \nu$	special-mixedulnu	1	0.05
$B \to X_u \ell \nu$	special-chargedulnu	1	0.05

The *offline analysis* part includes optimized cuts on feature variables in Section 5.3.1, the application of Monte Carlo corrections obtained from data Section 5.3.4, signal yield extraction (Section 5.3.4) and the classifier evaluation. Many parts of the offline analysis can be executed and visualized in a `Jupyter` notebook using a kernel from the `IPython` project[54], enabling also direct rapid prototyping. During offline analysis, many data objects are handled as `Pandas DataFrame` [55] and are saved in Hierarchical Data Format Version 5 `HDF5` [56].

The analysis framework developed in this thesis has successfully inspired other frameworks analyzing Belle data with fresh concepts, now being developed in the collaboration to streamline and automate analysis steps.

5.2. Available Data Sets and Skims

During the runtime of the Belle experiment, it recorded a total data set with a luminosity of 711 fb^{-1}on the $\Upsilon(4S)$ resonance. This data set contains roughly $772 \pm 11 \times 10^6$ B meson pairs. The collaboration provides supplementary sets of Monte Carlo, which are divided into different sets for specific decay types. The amount of events in a sample that corresponds to the amount of the total recorded luminosity of this decay type is referred to as a stream. The streams used in this study are listed in Table 5.1. The size of the data set multiplied by the given weight results in the expected quantities of that event type in the Belle data set. The *mixed* and *charged* samples contain only neutral and charged B mesons, respectively. The *charm* and *uds* samples contain charm and quark background, respectively. Since cross-feed from mis-reconstructed rare and inclusive decays might also have an influence on the tagging decision and contain a possible peaking background component, *rare* and $X_u \ell \nu$ samples are included as well. A pre-defined skim of the Belle collaboration, the *Hadron BJ skim* is applied on all samples. The selection criterions aim to remove background events from the data set, especially from beam gas scattering and two-photon events, to reduce computational resources and increase the sample purity. A minimum amount of two ECL clusters and three tracks originating in the vicinity of the interaction point is required. The visible energy of a reconstructed event must exceed 20% of the collision energy. Conditional requirements are introduced, to explicitly allow for J/ψ and $\Upsilon(2S)$ candidates.

Note that for 7 files of the data (Hadron BJ skim), which were recorded with the Belle Analysis Software Framework `BASF` version 2002-004016, no database information was available. Those files have been discarded. This affects a total number of 580'979 events, which is a negligible fraction of 0.019% of all data events (3'069'191'729) in recorded data.

5.3. Calibration of the Flavor Tagger

In this section, the calibration of the Deep Flavor Tagger on neutral and charged channels is described. For the calibration of the flavor tagging algorithms the following neutral and charged B meson decays were investigated

$$\begin{aligned}
&B^0 \to D^{*-}\pi^+ \\
&\qquad \Lsh\ \bar{D}^0\ \pi^-_{\text{slow}} \\
&\qquad\quad \Lsh\ K^+\ \pi^-
\end{aligned}$$

$$\begin{aligned}
&B^- \to D^0\ \pi^- \\
&\qquad \Lsh\ K^-\ \pi^+.
\end{aligned}$$

The charged conjugated decay chains are also included. All cuts and selections are performed identically for the charge conjugated decays. In parallel, the category based flavor tagger is also evaluated.

To obtain a better overview of the analysis, some the steps are described for the $B^0 \to D^{*-}\pi^+$ channel and the $B^- \to D^0\pi^-$ channel in full detail. For the neutral B meson channels, self tagging decays are used. The full Belle data set with the Hadron BJ skim applied is utilized for processing.

Besides that, the Belle collaboration provides several MC data sets, which are described in Section 5.2, including mixed and charged samples. Criteria for the selection of samples and the optimized cuts are discussed.

5.3.1. Variable Selection

The analysis does not only aim to reconstruct a channel with a high efficiency but to reconstruct it as cleanly as possible (highest purity region). This means adjusting the signal to noise ratio in a way to find the optimal tradeoff between purity and efficiency. For that reason, a figure of merit is used. For each cut the, signal (S) to noise (B) ratio

$$f = \frac{S}{\sqrt{S+B}} \tag{5.1}$$

is maximized. If the Monte Carlo distribution of cut variable has a large slope at the optimal value, the cut value is chosen to be slightly less stringent, to ensure not to cut in a region with high variance. The main reason for this is that an influence of a possible data vs. Monte Carlo agreement is kept at a minimum.

Since a fit for the final extraction is performed, reducing the background components is of greater importance than solely optimizing a figure of merit. Therefore, several cuts are made either to directly decrease the background components or increase the comparability between data and Monte Carlo samples.

Table 5.2.: Default track pre-cuts for each sample. No μID and eID pre-cuts are applied.

particle	KID	dr	$\lvert$d$z\rvert$	p_{CMS}
pion (slow)	< 0.4	< 2	< 4	
pion	< 0.4	< 2	< 4	> 0.5
kaon	> 0.6	< 2	< 4	

Pre-selection

After the initial reconstruction steps of a decay, the data is written out in a flat n-tuple format. For each track, multiple attributes including particle information and kinematic variables are extracted. To reduce the data size to a manageable amount, the pre-cuts described in the following are made. The pre-cuts are applied during the online processing; the computing expensive part is executed with `BASF2`. All other cuts and selections are applied on the level of the flat n-tuples.

The particle identification (PID) cuts with the kaon identification KID and criteria on track impact parameters, with the transverse distance to the interaction point dr and the distances to the interaction point on the z-axis $|$d$z|$ are listed in Table 5.2. Also cuts on the particle momentum in the $\Upsilon(4S)$ center of mass system p_{CMS} are applied. The value is adapted for the different particle hypotheses during the reconstruction.

Additional applied pre-cuts to higher level reconstructed particles are performed, depending on the PDG mass M_{PDG}, the reconstructed mass M_{Reco} and the released energy of a D^* meson, which corresponds to the slow pion momentum p_π^{slow} in the D^* center of mass system with

- $M_D^{\text{PDG}} - M_D^{\text{Reco}} \in [-0.16, 0.16]$ GeV
- and $p_\pi^{\text{slow}} \in [0.00, 0.075]$ GeV.

A D meson vertex fit was performed. No cuts are applied on the D meson vertex fit but candidates with a failed vertex fit were rejected.

Selection

At e^+e^- colliders, the initial state is well known. Therefore the beam energy E_{beam}, in terms of the total energy in the CMS system s is defined at

$$E_{\text{beam}} = \sqrt{s}. \tag{5.2}$$

This allows the definition of the energy difference ΔE to the energy of a reconstructed B meson with E_B with

$$\Delta E = E_B - \frac{1}{2} E_{\text{beam}}, \tag{5.3}$$

which is a powerful discrimination variable. The continuum suppression variables R_2 and $\cos(\theta_{th})$ are used to reduce large amount of the continuum background. They are defined in Section 2.3.2. To reduce wrongly selected particle candidates, loose cuts on the particle identification for electrons eID and muons μID are made. Also, cuts on the reconstructed D meson mass M_D or reconstructed D^* meson mass M_{D^*}

are applied. It is useful, to constrain the momentum of a reconstructed D meson $p_{D,CMS}$, as well.

For the $B^- \to D^0 \pi^-$ channel, the following selection was made:

- B meson: $\Delta E \in [-0.05, 0.05]$ GeV
- continuum: $R_2 < 0.4$, $\cos(\theta_{\text{th}}) < 0.8$
- first pion: eID < 0.9, μID < 0.5
- D meson: $M_D \in [1.85, 1.89]$ GeV.

For the $B^0 \to D^{*-}\pi^+$ channel, the following selection was made:

- B meson: $\Delta E \in [-0.05, 0.05]$ GeV
- continuum: $R_2 < 0.4$, $\cos(\theta_{\text{th}}) < 0.8$
- $M_{D^*} - M_D \in [0.143, 0.146]$ GeV
- slow pion: eID < 0.9, μID < 0.5
- D meson: $p_{D^*,CMS} \in [2.0, 2.5]$ GeV.

5.3.2. Fit Variable

For signal yield extraction, usually a fit variable with the best expected discrimination power is chosen. Ideally this variable is uncorrelated with the discrimination variables of previous signal selection steps. These steps can include cuts, multivariate methods or a best candidate selection. Furthermore, the statistical composition of signal and background distributions have to be understood very well for the reconstructed decay channel.

A variable that allows a clean kinematical separation of signal and background candidates is the *beam constrained mass* M_{bc}.

Instead of using the invariant mass of a particle as a discrimination variable, prior knowledge about the decay structure can be utilized. In case of the $\Upsilon(4S)$ resonance, the decay products are usually two B mesons. A reconstructed B meson with an energy E_B and a reconstructed three-momentum p_B is replaced with the beam constrained energy E_{beam} to construct the beam constrained mass with

$$M_{\text{bc}} = \sqrt{\frac{E_{\text{beam}}^2}{4} - p_B^2}. \tag{5.4}$$

Mis-reconstructed B mesons often have a shift in momentum to the lower tail of the distribution.

5.3.3. Background Composition

The background is composed of a peaking and a non peaking component. The non peaking component consists mostly of wrongly combined decay candidates, which results in a lower beam constraint mass than signal candidates. In contrast, the majority of the peaking component originates from wrongly reconstructed particles

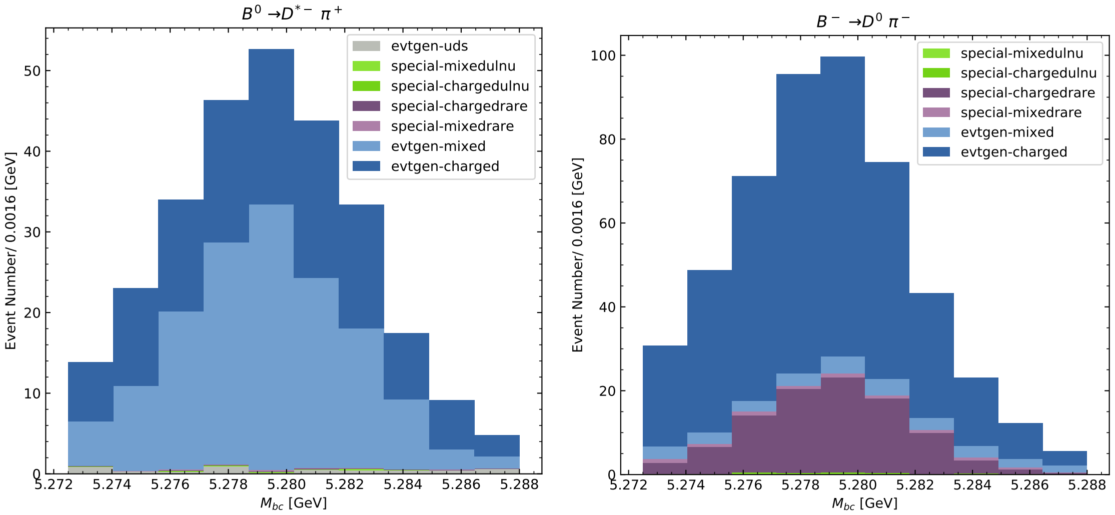

Figure 5.2.: Peaking background components in the signal window of the beam constraint mass M_{bc}. The neutral B meson channel is shown on the left hand side, the charged B meson channel shown in the right hand side. Components, which have a flat shape in this window and can be clearly determined in lower energy part of the distribution, are not shown.

from other B meson decays which do not belong to the reconstructed signal channel, but match kinematically. In Figure 5.2 the components, which contain charged and neutral B meson decays contribute most to the peaking background. For the peaking window for the components is defined in the range of $M_{\mathrm{bc}} \in [5.273, 5.288]$ GeV.

The $\Upsilon(4S)$ resonance decays into two B mesons. Here, the definition of the wrongly matched decay channel is ambiguous, since in the usual wrongly matched case, products of both B meson decays are mis-matched. For the definition of a background component in the decay, the reconstructed B meson decay component with the most matches is chosen. The top 90% of the peaking background components for the $B^0 \to D^{*-}\pi^+$ channel are listed in Table 5.3. Although for this channel, these components only represent a tiny fraction of 2.8% of all events, for the determination of the systematic uncertainties, a characterization of this background is crucial. Identifying the peaking components helps in the choice of selection variables for their suppression and allows for a more accurate determination of the systematic uncertainties. Wrongly matched signal events can have a significant influence on the tagging decision, since potential characteristic signal particles are now assigned to the tag side. With a fraction of 7% of all background events and a fraction of 0.2% of all events this is negligible for this channel. The components for the channel $B^- \to D^0\pi^-$ are listed in Table 5.4.

5.3.4. Monte Carlo Procedure

Monte Carlo is generated according to a physics model, and new measurements have to be integrated into the parameter space used by the models. Generated Monte Carlo can therefore only mimic data to a certain extent, depending of the detailed level of the model, the knowledge of the parameters and the complexity of the simulations. The majority of the Monte Carlo samples are produced with the

Table 5.3.: Main contribution of B meson channels to the peaking background component for the signal channel $B^0 \to D^{*-}\pi^+$ with 89% of all contributions. The peaking component for this channel is only a fraction of less than 3% of the total events and a fraction of approximately 28% of all background events. All fractions are determined on Monte Carlo.

decay channel	**fraction** [%]
$B^+ \to \bar{D}^*(2007)^0\ \pi^+$	16.7
$B^- \to D^*(2007)^0\ \pi^-$	15.8
$B^0 \to D^*(2010)^-\ \pi^+$	10.8
$\bar{B}^0 \to D^*(2010)^+\ \pi^-$	10.4
$B^0 \to D^-\ \pi^+$	8.1
$\bar{B}^0 \to D^+\ \pi^-$	7.4
$\bar{B}^0 \to D^*(2010)^+\ K^-$	6.4
$B^0 \to D^*(2010)^-\ K^+$	5.8
$B^- \to D^0\ \rho^-$	4.2
$B^+ \to \bar{D}^0\ \rho^+$	3.6

Table 5.4.: Main contribution of B meson channels to the peaking background component for the signal channel $B^- \to D^0\pi^-$ with 81% of all contributions. The peaking component for this channel is a fraction of less than 1% of the total events and less than 7% than all background events. All fractions are determined on Monte Carlo.

decay channel	**fraction** [%]
$B^+ \to \bar{D}^0\ K^+$	17.9
$B^- \to D^0\ K^-$	17.0
$B^+ \to \bar{D}^*(2007)^0\ \pi^+$	8.7
$B^- \to D^*(2007)^0\ \pi^-$	7.8
$B^+ \to K^*(1430)^0\ \pi^+$	5.3
$B^- \to \bar{K}^*(1430)^0\ \pi^-$	5.2
$B^- \to D^0\ \pi^-$	4.6
$B^+ \to D^0\ \pi^+$	4.5
$B^+ \to K^*(892)^0\ \pi^+$	2.7
$B^- \to \bar{K}^*(892)^0\ \pi^-$	2.6
$B^- \to D^0\ \rho^-$	2.2
$B^+ \to \bar{D}^0\ \rho^+$	2.1

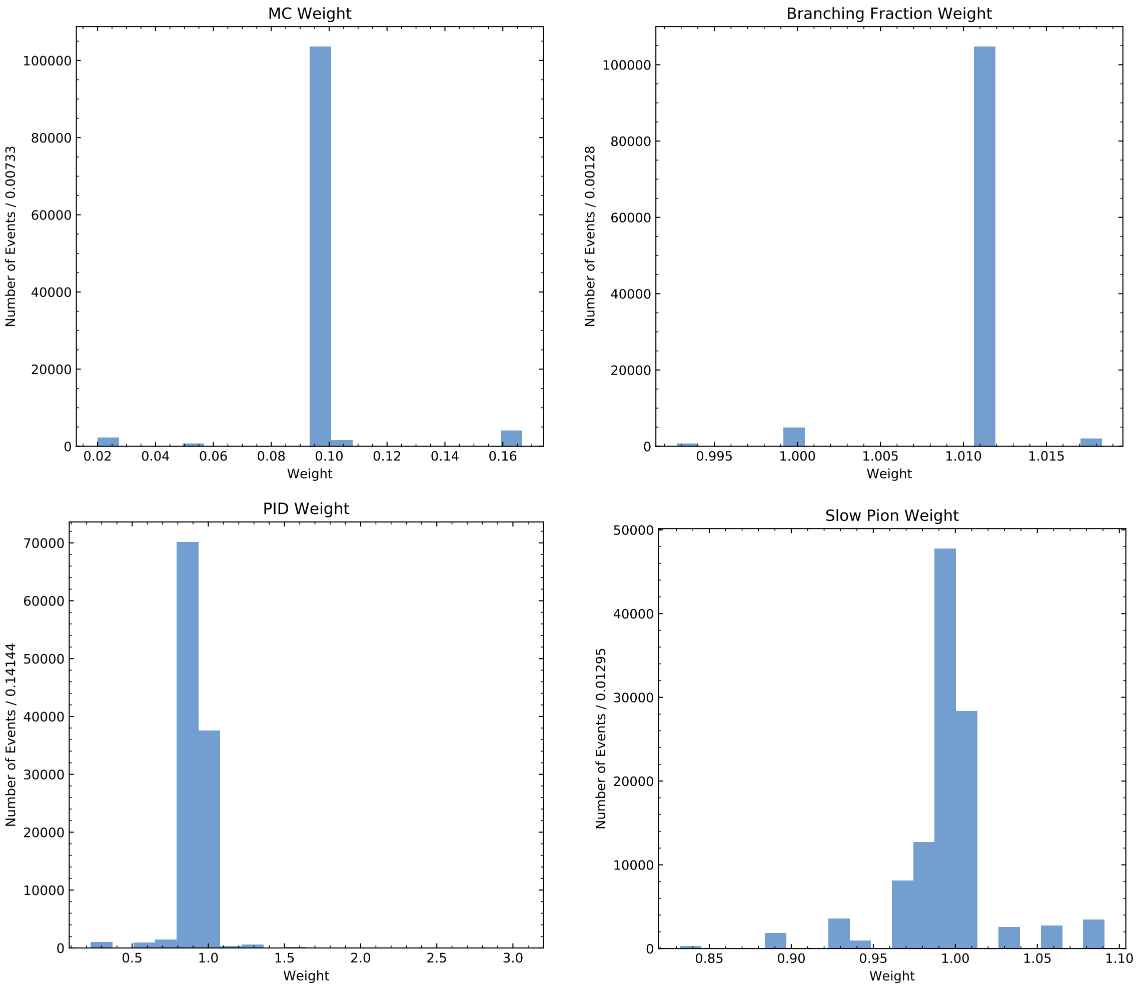

Figure 5.3.: Different weights for reconstructed events of the decay $B^0 \to D^{*-}\pi^+$ on Monte Carlo. On the upper left, the MC weights obtained by the streams are shown. On the upper right, the weights due to branching fraction re-weighting are visualized. On the lower left, the weights due to particle identification corrections and on the lower right, the weights due to slow pion corrections are illustrated.

Monte Carlo Generator `EvtGen` [57]. The detector simulation including the material effects caused by traversing particles are simulated with `GEANT3` [58].

The Belle data was recorded over a decade ago. However, it has a high information content and is still valuable today. Even today new analyses are being performed on this data set. While measurements and analytic procedures make progress over time, it is often not feasible to redo Monte Carlo generation due to the high computing demand. Sometimes it is even not possible to change some specific aspects of the Monte Carlo data without re-implementing and re-calibrating major parts. The MC weights without re-weighting for the channel $B^0 \to D^{*-}\pi^+$ are shown in the upper left of Figure 5.3.

Branching Fraction Re-Weighting

New branching fractions are measured, other values are corrected by new measurements or updated because more measurements can be taken into account when

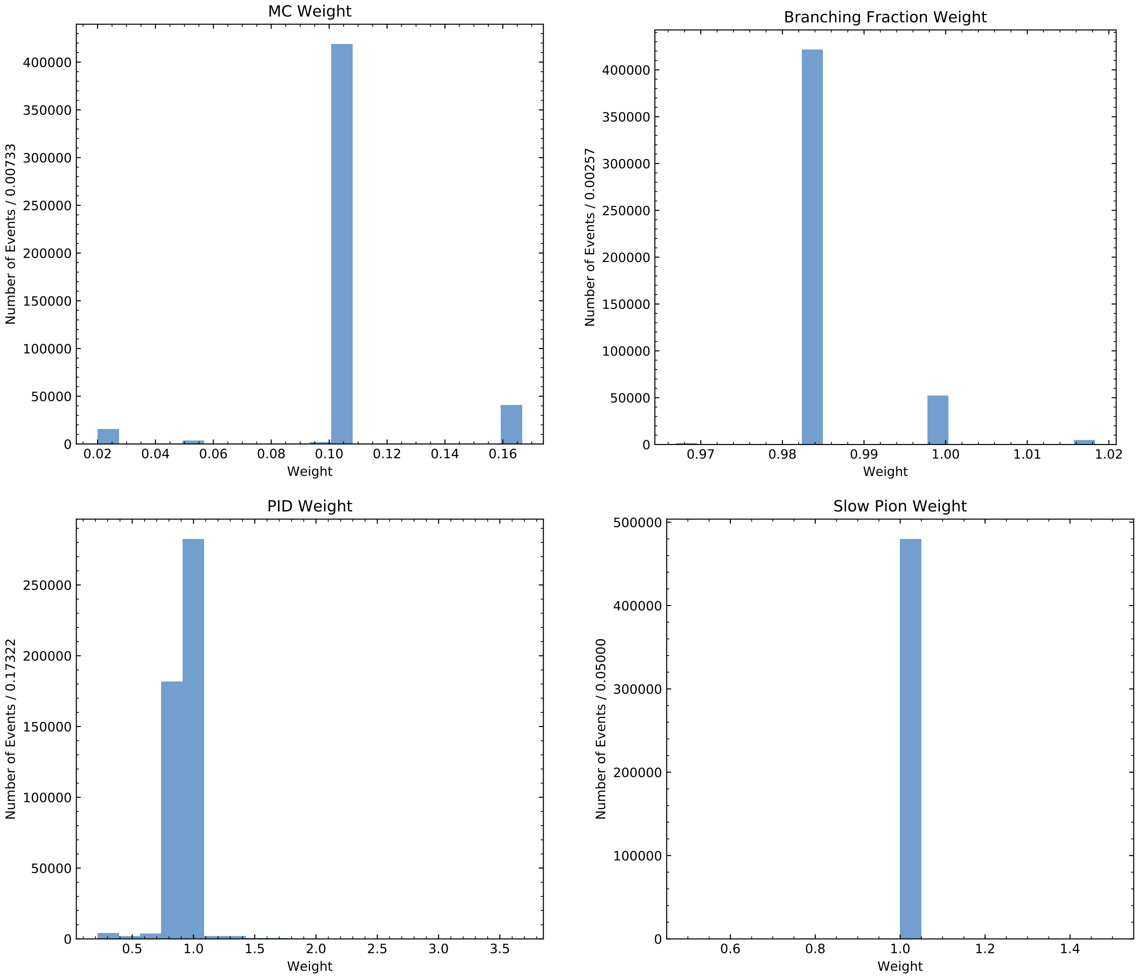

Figure 5.4.: Different weights for reconstructed events of the decay $B^- \to D^0\pi^-$ on Monte Carlo. On the upper left, the MC weights obtained by the streams are shown. On the upper right, the weights due to branching fraction re-weighting are visualized. On the lower left, the weights due to particle identification corrections and on the lower right, the weights due to slow pion corrections are illustrated.

building the average. This analysis relies on Monte Carlo samples, where some have been produced over a decade ago (2008), therefore new findings from high energy physics have to be integrated. The different weights of branching fractions have a significant impact to the shape of the probability density functions, obtained by reconstruction of decays and features in Monte Carlo samples. The branching fractions for the signal channels used were updated for each sub-channel. For the decay $B^0 \to D^{*-}\pi^+$ the corrective weights for the branching fractions are shown in the upper right of Figure 5.3.

Particle Identification

Particle identification plays a major role in mapping a track to a certain particle type. At Belle, a combined likelihood ratio over multiple detectors for muons versus electrons $\mathcal{L}(\mu, e)$ and kaons versus pions $\mathcal{L}(K, \pi)$ are determined on Monte Carlo and data. Performance studies of decays have been performed for leptonic IDs [59, 60] and hadronic IDs [61]. The data of these studies have been updated continuously until 2009 to include data from newer experiments. In these studies, the yields were determined with and without particle identification cuts applied. With those studies, efficiency and mis-identification rate for a specific cut were determined. The parameters have been measured with respect to a small set of variables, including momentum and angular dependencies. For the hadronic measurements, the channel $D^{*+} \to D^0\pi^+$ was used. The systematic uncertainties and corrections for particle identification applied in this thesis were taken from [61]. The corrective weights for particle identification for the channel $B^0 \to D^{*-}\pi^+$ can be found in the lower left of Figure 5.3.

Tracking

The uncertainty of the tracking efficiency was investigated in [62]. A study for data and Monte Carlo disagreement on D^* decays in experiment 7-23 data and 7-19b MC has been performed. Here, the track finding efficiencies η_{Data} and η_{MC} on data and MC, respectively, are compared and a ratio $r = \frac{\eta_{\text{Data}}}{\eta_{\text{MC}}}$ is calculated.

An uncertainty on the tracking efficiency asymmetry $a = r - 1$ is obtained with $a = (0.35 \pm 0.82 \pm 0.20)\%$ (stat, sys) which is compatible with zero within the uncertainties. In agreement with the standard procedure of the Belle collaboration, an uncertainty of 0.35% for each track is used. Since it is assumed that these efficiencies are maximally correlated, the systematic for each track in the signal channel is added linearly. Additionally for slow pion candidates, a separate correction is calculated. These corrections are shown for the example channel in the lower right of Figure 5.3.

Curler Clone Finder

In a small number of cases, a track is recognized multiple times by the track reconstruction algorithm. A large source of these clones are so-called curlers, which traverse similar regions in the detector multiple times. They are characterized by similar angular helix parameters and a similar transverse momentum. On Monte Carlo, a cut on specific set of curler variables on the tag side reduces a cross-feed induced bias between the true efficiency and the measured efficiency on the tag side. Unfortunately, this selection could not be applied on the rest of event for the category based approach due to limitations of the version of the algorithm implementation.

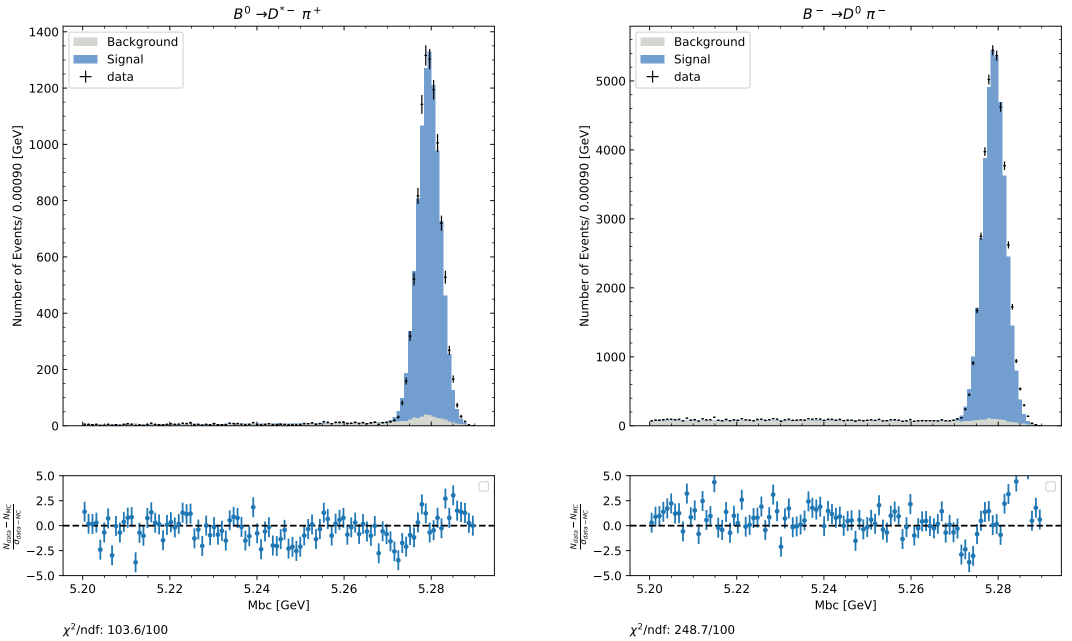

Figure 5.5.: Reconstructed events in the specific signal channels for data and all available streams of Monte Carlo. The yields are not fitted in this comparison. Signal and Background events are marked in the Monte Carlo samples.

Signal Yield Extraction

In Section 5.3.2, the beam constraint mass as a fitting variable is introduced, which will be used for the extraction of the signal yield. As a comparison, the distribution of the fitting variable is shown for the different channels in Figure 5.5.

The fit is subdivided into three different parts, one signal component, a non-peaking background component and a peaking background component. For the signal component, a *Crystal Ball* shape function f_{CB} is chosen. It is composed of a Gaussian component with an exponential tail. Most of the well reconstructed B mesons end up in the Gaussian part, centered around the invariant mass of the B meson. The tail catches mostly candidates with missing particles, for instance π^0 components. Usually, the limited calorimeter resolution has the effect, that not all photons are detected which are required for the reconstruction of the particle. With the parametrization in the fit variable m and the parameters α, m_{CB} and σ_{CB}, the Crystal Ball function is given by

$$f_{\mathrm{CB}}(m;\alpha,n,m_{\mathrm{CB}},\sigma_{\mathrm{CB}}) = N \begin{cases} e^{-\frac{(m-m_{\mathrm{CB}})^2}{2\sigma_{\mathrm{CB}}^2}}, & \text{for } \frac{m-m_{\mathrm{CB}}}{\sigma_{\mathrm{CB}}} > -\alpha \\ A(n,\alpha)\cdot(B(n,\alpha)-\frac{m-m_{\mathrm{CB}}}{\sigma_{\mathrm{CB}}})^{-n}, & \text{for } \frac{m-m_{\mathrm{CB}}}{\sigma_{\mathrm{CB}}} \leq -\alpha \end{cases} \tag{5.5}$$

with $A(n,\alpha) = \left(\frac{n}{|\alpha|}\right)^n \cdot e^{-\frac{|\alpha|^2}{2}}$ and $B(n,\alpha) = \frac{n}{|\alpha|} - |\alpha|$ and a normalization factor N.

The non-peaking background component is parametrized with an *ARGUS* function f_{ARG}. The endpoint m_{ARG} is fixed to 5.2889 GeV, the maximum reachable value for

M_{bc}, since it is limited by the beam energy.

The ARGUS function is parametrized with the parameters m_{ARG}, c, p as

$$f_{\mathrm{ARGUS}}(m; m_{ARG}, c, p) = N \cdot m \cdot \left[1 - \left(\frac{m}{m_{\mathrm{ARG}}}\right)^2\right]^p \cdot \exp\left[c \cdot \left(1 - \left(\frac{m}{m_{\mathrm{ARG}}}\right)^2\right)\right]. \tag{5.6}$$

The peaking background component is represented by a *Gaussian* shape function, with the width μ and the variance σ_G^2

$$f_{\mathrm{gaussian}}(m; \mu, \sigma_G) = \frac{1}{\sigma_G\sqrt{2\pi}} e^{-\frac{1}{2}((m-\mu)/\sigma_G)^2}. \tag{5.7}$$

Unbinned Extended Maximum Likelihood Fit

The recorded data represents a subsample of the underlying distribution. The goal is to adapt the parameters of a model, so that it fits best to a given set of n data points. The signal yield is extracted with an unbinned extended maximum likelihood fit. The procedure is described as follows, for more details [63] is recommended.

For a given probability density function f with the fit variable x and the parameters θ the likelihood L is given as the joint probability density function over all data points i

$$L(\theta) = \prod_{i=1}^{n} f(x_i; \theta), \tag{5.8}$$

which is to be maximized. For ease of calculation and to allow for a better numerical stability, usually the log likelihood function $\log(L)$ is used.

Since this method would only allow the determination of relative fractions, the extended maximum likelihood is used. In this case, the frequency of observed events is treated as a Poisson distributed random variable around ν, and the maximum likelihood term extends to

$$L(\nu, \theta) = \frac{\nu^n}{n!} e^{-\nu} \prod_{i=1}^{n} f(x_i; \theta). \tag{5.9}$$

The maximum (or minimum by multiplying $\log(L)$ by -1) can be found by a parameter adjustment with a gradient descent algorithm.

Implementation of the Fit

In this thesis, the algorithm `MIGRAD` from the `MINUIT` package [64] which is provided by `RooFit` [65] has been used. The algorithm solves the optimization problem to find a minimum value of a function in a vast parameter space with a gradient descent procedure. By the procedure itself, there is no guarantee that the global minimum is found. In case covariance matrices are required, they are determined by the optimization algorithm `HESSE`, using results of the processing steps of `MIGRAD`. Since the analysis framework of this thesis is implemented in `Python`, a wrapper around `RooFit` is utilized [66]. Automated checks for errors and failed fits are performed.

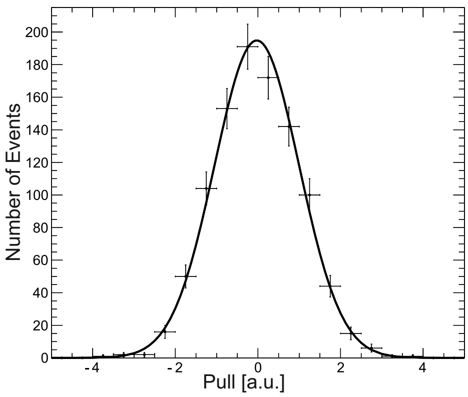

Figure 5.6.: Pull distribution of yield for the signal component N_{sig} for 1000 fits, sampled from a poisson distribution based on stream MC with a mean of $\mu = (-2.74 \pm 3.27) \cdot 10^{-2}$ and standard deviation of $\sigma = (1.023 \pm 0.023)$.

The fitting procedure is structured in three steps. At first, the non-peaking f_{ARGUS} component and the peaking gaussian component is fitted on background MC events separately. All parameters are floating. Then the signal shape of the Crystal Ball function is determined by fitting the Crystal Ball function on signal MC events only. The parameters n and α of the Crystal Ball function, as well as the norm N and c of the ARGUS function are fixed at the value obtained in the pre-fits. Then the final fit for the remaining components and normalizations are performed on data to extract the desired quantities.

Validation of the Fit

To validate the stability of the fit *Monte Carlo toy studies* are performed. From the underlying distribution, different sub-samples according to the number of events of a Poisson statistics are drawn. The number of events is fitted, and the pull distribution $x_{\text{pull},i}$ is calculated with

$$x_{\text{pull},i} = \frac{N_i - N_{\text{true}}}{\sigma_i} \tag{5.10}$$

where N_i is the number of fitted events, N_{true} is the true number of events of the drawn distribution and σ_i is the uncertainty of the fit. Subsequently, the signal extraction fit to evaluate the pull distribution is performed 1000 times.

For a perfectly working fitting procedure with accurate uncertainty estimation, a Gaussian distribution with $\mu = 0$ and $\sigma = 1$ is expected. The pull distribution The results for the applied fit approach for the $B^0 \to D^{*-}\pi^+$ channel are shown in Figure 5.6 with a mean $\mu = (-2.74 \pm 3.27) \cdot 10^{-2}$ with a standard deviation $\sigma = (1.023 \pm 0.023)$.

To check the stability of the fit procedure, the final fit is performed on all 6 MC streams separately. The results are shown in Figure 5.7.

5.3.5. Systematic Uncertainties

To be able to precisely measure a quantity, the systematics of the measurements have to be understood. There are different types of systematic effects, which affect

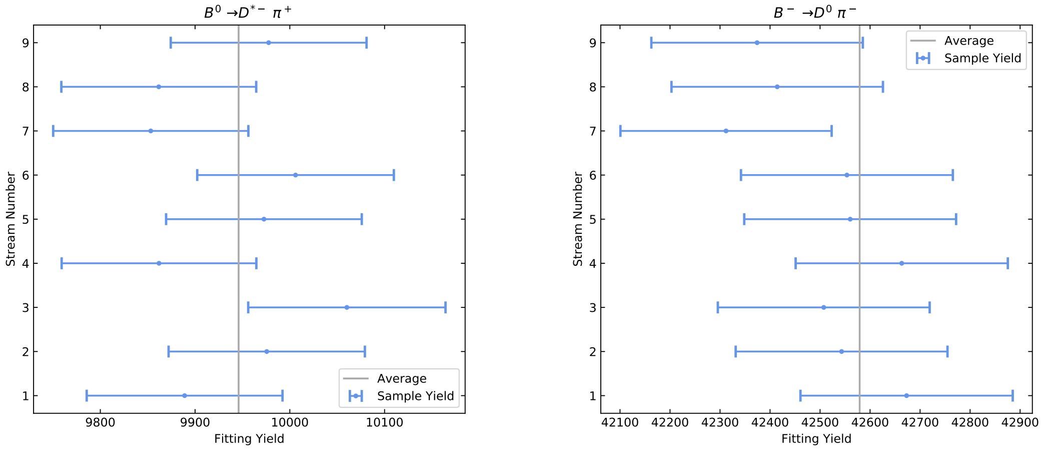

Figure 5.7.: Fitted signal yield on 9 of 10 available *charged* and *mixed* Monte Carlo streams. The stream zero is used for fixing the background yield for these cross checks. For the final fits, the parameters are fixed on the combined information of all streams.

the number of reconstructed B meson pairs and measured B meson yields in the specific channels. The following systematic uncertainties are considered:

- A possible *fit bias*, which is investigated in Section 5.3.4,
- statistical uncertainties in the *Monte Carlo background shape*,
- *particle identification efficiency*,
- the *number of B meson pairs*,
- the uncertainty on the *tracking efficiency* and
- the MC statistics for efficiencies (negligible for relative fractions on a single channel, negligible with sufficiently high signal MC statistics).

The uncertainties for the particle identification efficiency are such an important source of information for the reconstruction, that there were several studies performed, see Section 5.3.4. The uncertainties for all PID weights are propagated to a single number. For each systematic component i, a relative systematic uncertainty is determined. The Monte Carlo weights w_{ij} are varied independently according to the specific systematic uncertainty σ_{ij} individually to obtain the varied weights $\tilde{w}_{ij}$

$$\tilde{w}_{ij} = w_{ij} \pm \sigma_{ij}. \tag{5.11}$$

The fit is performed repeatedly for the upward and downward fluctuations (±), in the bins of the flavor tagger k. The obtained yield for the upward and downward fluctuations is $n_{i\pm}^{(k)}$. The returned yield is composed of a central value $\tilde{n}_i^{(k)}$ and an upward or downward fluctuation σ_{i+} or σ_{i-}, respectively with

$$n_{i+}^{(k)} = \tilde{n}_i^{(k)} + \sigma_{i+} \tag{5.12}$$
$$n_{i-}^{(k)} = \tilde{n}_i^{(k)} - \sigma_{i-}. \tag{5.13}$$

Table 5.5.: Systematic uncertainties for the channel $B^0 \to D^{*-}\pi^+$ in relative values for the effective tagging efficiency $\mathcal{Q}$. The systematic uncertainties for the signal shape cancel out of the fractions and will be neglected.

Efficiency Systematics		[%]
	χ_d	0.700637
Signal	Branching Fraction	0.007768
	Number of B Meson Pairs	0.009490
	Particle Identification Correction	0.020724
	Track Reconstruction Efficiency	0.009413
Peaking Background	Branching Fraction	0.115891
	Number of B Meson Pairs	0.012368
	Particle Identification Correction	0.008294
	Track Reconstruction Efficiency	0.010669

For each component, the absolute uncertainty is extracted from the fitted yields and symmetrized

$$\sigma_i^{n,\mathrm{sym}(k)} = \frac{n_{i+}^{(k)} - n_{i-}^{(k)}}{2}. \tag{5.14}$$

Note that the central value is implicitly absorbed by using the difference between both yields to obtain the symmetrized uncertainty $\sigma_i^{n,\mathrm{sym}}$. Correlation matrices between the fitted yields are constructed via the outer product

$$\mathrm{cov}(n_i) = \sigma_i^{n,\mathrm{sym}} \otimes \sigma_i^{n,\mathrm{sym}}. \tag{5.15}$$

For each component of the corresponding uncertainties, a small possible higher order correlation is neglected and a combined covariance matrix is determined with

$$\mathrm{cov}(n) = \sum_i \mathrm{cov}(n_i) \tag{5.16}$$

assuming a 100% correlation for the individual uncertainty types.

The determination of the effective tagging efficiency, defined in Section 5.4, only takes fractions or sums of fractions of yields into account. The uncertainties from the signal component only play a minor role. For every uncertainty component, fits on data are performed using the varied background templates. The fits are performed in bins of the classifier output. The obtained uncertainties for the effective tagging efficiency are shown in Table 5.5 and in Table 5.6. The uncertainties for the category based approach are shown in Table A.5 and in Table A.6.

5.4. Calibration Factors

Even though tremendous efforts by several groups are undertaken to achieve a better understanding of the underlying data and come closer to data/ Monte Carlo agreement, there are still small discrepancies remaining. Since a multivariate method heavily relies on correlations between multiple variables, differences can result in unequal behavior between data and Monte Carlo. The current state of the art agreement is still far from perfect.

Table 5.6.: Systematic uncertainties for the channel $B^- \to D^0\pi^-$ in relative values for the effective tagging efficiency $\mathcal{Q}$. The systematic uncertainties for the signal shape cancel out of the fractions and will be neglected.

Efficiency Systematics		[%]
Signal	Branching Fraction	0.002293
	Number of B Meson Pairs	0.000417
	Particle Identification Correction	0.001204
	Track Reconstruction Efficiency	0.001377
Peaking Background	Branching Fraction	0.026406
	Number of B Meson Pairs	0.001037
	Particle Identification Correction	0.003156
	Track Reconstruction Efficiency	0.001046

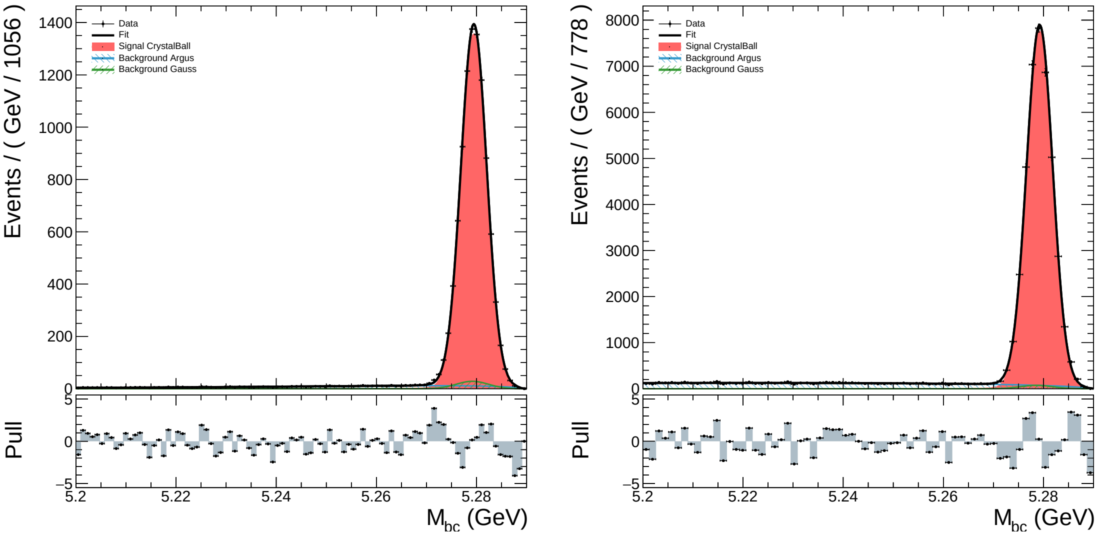

Figure 5.8.: Fits on data for $B^0 \to D^{*-}\pi^+$ decay on the left hand side and for $B^- \to D^0\pi^-$ on the right hand side. The signal and background components are displayed separately.

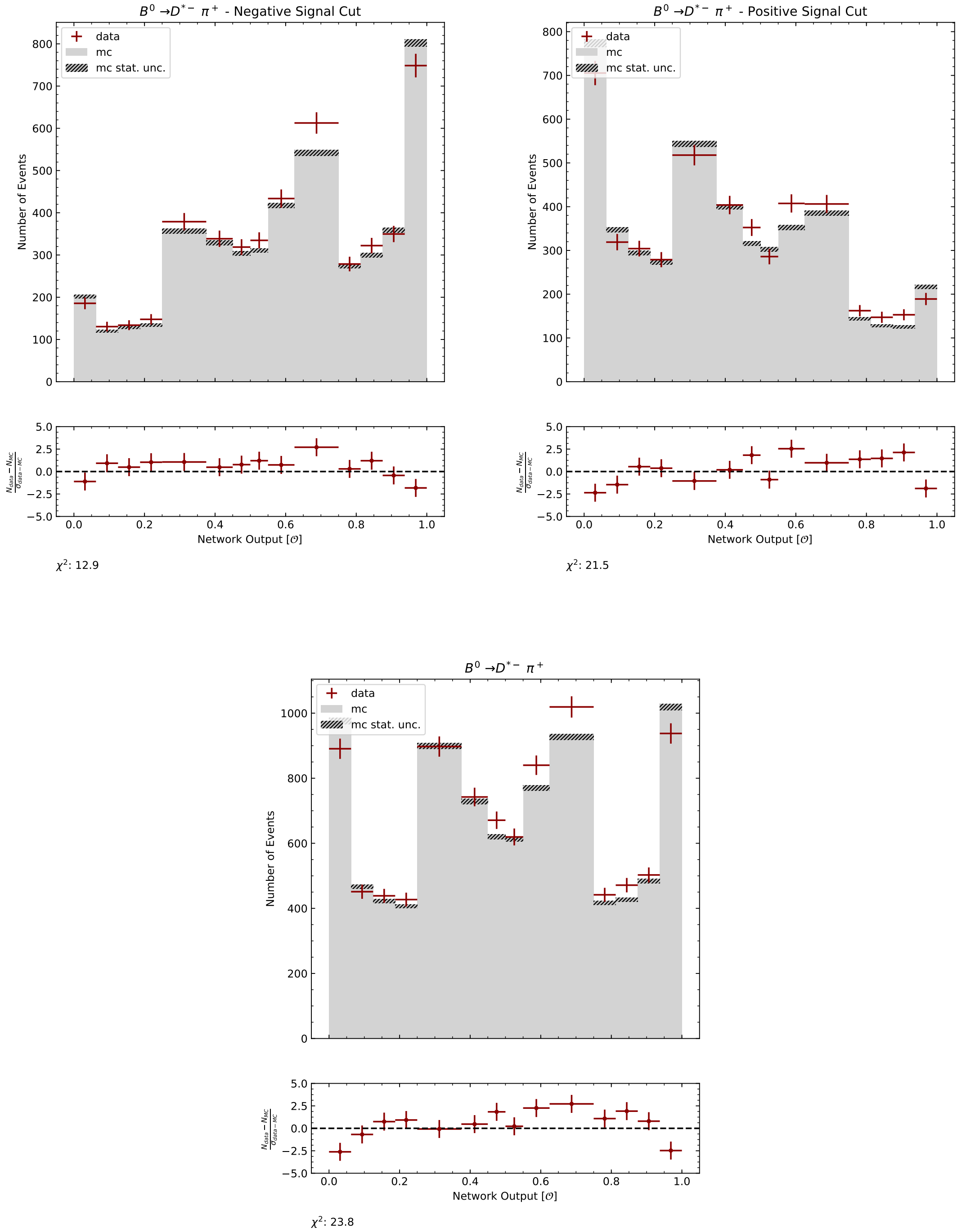

Figure 5.9.: Histograms of the classifier output of the Deep Flavor Tagger for the $B^0 \to D^{*-}\pi^+$ decay, separated by the charge of the reconstructed signal B meson (upper left and upper right). The Monte Carlo shape is shown in gray and the fitted yields of the data sample are shown in red in each bin. The classifier output for both charges is shown in the lower half.

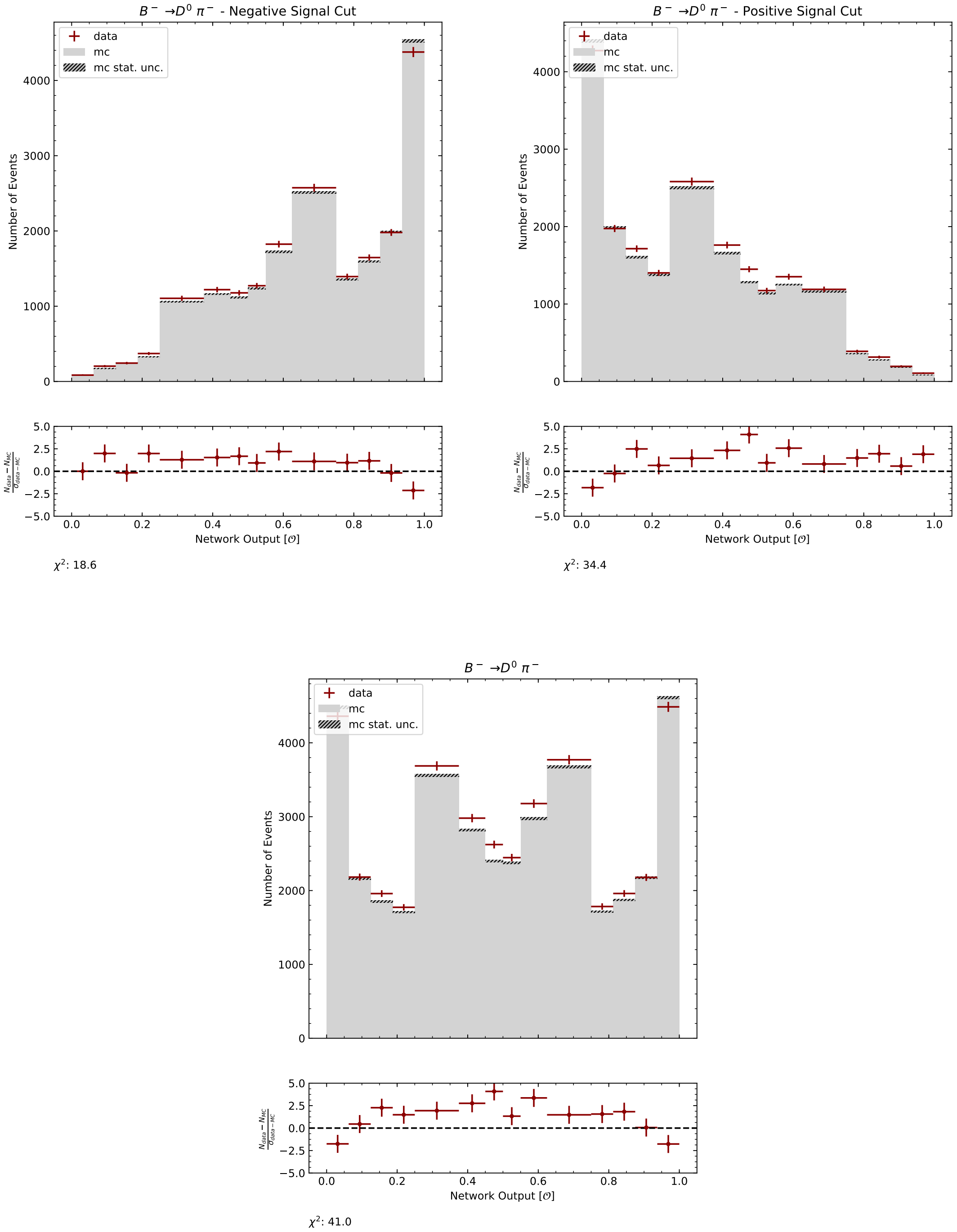

Figure 5.10.: Histograms of the classifier output of the Deep Flavor Tagger for the $B^- \to D^0\pi^-$ decay, separated by the charge of the reconstructed signal B meson (upper left and upper right). The Monte Carlo shape is shown as a histogram, the fitted yields of the data sample are shown in the corresponding bins. The classifier output for both charges is shown in the lower half.

So-called self tagging decays allow probability estimations for the flavor of the accompanying B meson. They can be used to validate and calibrate the tagging decisions of the algorithm, justifying usage of more powerful multivariate methods. In the following section it will be shown how the wrong tag fraction can be evaluated directly on data.

The e^+e^- collider runs at an energy where it usually produces $\Upsilon(4S)$, which almost always decay into two B mesons. In the case of neutral B mesons, it is not clear which pairs will exist at the time of the decay. Even though they are produced in pairs with opposite-flavors and evolve coherently, due to mixing and different decay times, all combinations of $B^0\bar{B}^0$, $B^0B^0$and $\bar{B}^0\bar{B}^0$can be measured in one event.

For neutral B_d^0 mesons, CP violation in mixing is negligible, as described in Section 3.2. Therefore, it is possible to describe the number of measured B and $\overline{B}$ mesons, denoted by N_B and $N_{\overline{B}}$, respectively as a function of the measured flavors. It can be integrated over the complete decay time phase space. The admixture of obtaining $\tilde{N}_B$ or $\tilde{N}_{\overline{B}}$ events on the tag side of an event, when measuring N_B or $N_{\overline{B}}$ events on the signal side, respectively, can be described with

$$\begin{pmatrix} 1-\chi_d & \chi_d \\ \chi_d & 1-\chi_d \end{pmatrix} \begin{pmatrix} N_B \\ N_{\overline{B}} \end{pmatrix} = \begin{pmatrix} \tilde{N}_B \\ \tilde{N}_{\overline{B}} \end{pmatrix}. \tag{5.17}$$

The time integrated mixing fraction is defined in absence of CP violation as [27]

$$\chi_d = \frac{x_d^2 + y_d^2}{2(x_d^2+1)} \tag{5.18}$$

with $x_d = \frac{\Delta m_d}{\Gamma_d}$ and $y_d = \frac{\Delta\Gamma_d}{2\Gamma_d}$. A combined measurement [27] under the assumption $\Delta\Gamma_d = 0$ and no CP violation in mixing ($|p/q| = 1$) is

$$\chi_d = 0.1860 \pm 0.0011. \tag{5.19}$$

5.4.1. Metrics for Flavor Tagging

For future analyses, the major area of application for a flavor tagger will be the measurement of flavor asymmetries. Therefore it will be of special interest to count the number of B mesons that changed their initial flavor and the number of B mesons that did not change their flavor.

Since the flavor estimation on the tag side by a flavor tagger corresponds to a probability, and not the complete information of the decay is available to the tagging algorithm, the selected events have an imperfect purity by definition. To quantify the performance of the tagging algorithm, the wrong tag fraction for B mesons $\widetilde{w}_B$ and $\overline{B}$ mesons $\tilde{w}_{\overline{B}}$ is introduced. The total number of true B or $\overline{B}$ events of a sample, N_B^{truth} or $N_{\overline{B}}^{\text{truth}}$ respectively, differ from the corresponding measured quantities N_B^{measured} and $N_{\overline{B}}^{\text{measured}}$ with

$$N_B^{\text{measured}} = \tilde{w}_B N_B^{\text{truth}} + (1-\tilde{w}_{\overline{B}}) N_{\overline{B}}^{\text{truth}} \tag{5.20}$$

$$N_{\overline{B}}^{\text{measured}} = \tilde{w}_{\overline{B}} N_{\overline{B}}^{\text{truth}} + (1-\tilde{w}_B) N_B^{\text{truth}}. \tag{5.21}$$

Usually, the wrong tag fractions are expressed as the average $\tilde{w} = \frac{1}{2}(\tilde{w}_B + \tilde{w}_{\overline{B}})$ and the difference $\Delta\tilde{w} = \frac{1}{2}(\tilde{w}_B - \tilde{w}_{\overline{B}})$ between both flavors.

The coherence of the $\Upsilon(4S)$ decay allows a categorization of the B meson flavors in terms of relative decay times without neglecting necessary information. Instead of counting the number of B mesons on tag-side or signal-side with a specific flavor, the kind of B meson pairs in an event can be used as a property. An event with a tag side B meson with the same or opposite-flavor as the signal side B meson can be described as same-flavor (SF) or opposite-flavor (OF) event, respectively. The measured asymmetry $\mathcal{A}$ is given by

$$\mathcal{A}(\Delta t) = \frac{N_{\mathrm{SF}}(\Delta t) - N_{\mathrm{OF}}(\Delta t)}{N_{\mathrm{SF}}(\Delta t) + N_{\mathrm{OF}}(\Delta t)}, \tag{5.22}$$

where Δt is the relative time difference between both decays.

With the previously used assumptions regarding B_d^0 mixing and with Equation (3.47), the probability for obtaining a same-flavor $\mathcal{P}^{\mathrm{SF}}$ or opposite-flavor event $\mathcal{P}^{\mathrm{OF}}$ for neutral B mesons can be obtained with

$$\mathcal{P}(\Delta t)^{\mathrm{SF/OF}} = \frac{1}{4\tau} e^{-|\Delta t|/\tau}(1 \pm (1-2w)\cos(\Delta m \Delta t)), \tag{5.23}$$

taking into account a wrong tag fraction from flavor tagging. Note, that this is a redefinition of the wrong tag fraction w, since it is now a measure for the wrongly tagged same-flavor and opposite-flavor events.

The observed mixed fraction χ_{obs} can be obtained by integrating over the full range of Δt

$$\chi_{\mathrm{obs}} = \frac{\int\limits_{-\infty}^{\infty} \mathcal{P}^{\mathrm{SF}}(\Delta t)\mathrm{d}\Delta t}{\int\limits_{-\infty}^{\infty} \mathcal{P}^{\mathrm{OF}}(\Delta t)\mathrm{d}\Delta t + \int\limits_{-\infty}^{\infty} \mathcal{P}^{\mathrm{SF}}(\Delta t)\mathrm{d}\Delta t} \tag{5.24}$$

for which one obtains

$$\chi_{\mathrm{obs}} = \chi_d + (1-2\chi_d)w. \tag{5.25}$$

To measure χ_{obs}, the fractions of the same-flavor and opposite-flavor B meson

$$\chi_{\mathrm{obs}} = \frac{N_{\mathrm{SF}}}{N_{\mathrm{SF}} + N_{\mathrm{OF}}} \tag{5.26}$$

have to be determined.

By including the determination of χ_d from other measurements as mentioned above, the wrong tag fraction w can be obtained with

$$w = \frac{\chi_{\mathrm{obs}} - \chi_d}{1 - 2\chi_d}. \tag{5.27}$$

The relation between a measured asymmetry $\mathcal{A}_{\mathrm{measured}}$ and actual asymmetry $\mathcal{A}$ can be expressed in terms of the wrong tag fraction with

$$\mathcal{A}_{\mathrm{measured}}(\Delta t) = (1-2w)\mathcal{A}(\Delta t). \tag{5.28}$$

Therefore it is reasonable to introduce the dilution factor $D = 1 - 2w$. Furthermore, since the relation between purity and efficiency of a classifier is not constant for different ranges of the classifier output, it is useful to evaluate and calibrate this metric in several bins.

Due to historical reasons and comparability between studies, 14 classifier bins l_i defined by the following 15 limits have been chosen:

$$\begin{aligned}\text{limits}(l_i) = {} & 0.0, 0.0625, 0.125, 0.1875, 0.25, 0.375, 0.45, 0.5, \\ & 0.55, 0.625, 0.75, 0.8125, 0.875, 0.9375, 1.0 \end{aligned} \tag{5.29}$$

For previous flavor tagging algorithm implementations, these bins were chosen since the majority of events of different categories of the conventional method accumulated in these bin regions. Depending on the B meson flavor of the signal side, the first or second half of the bins are defined as opposite and same-flavor bins, respectively. A value of one means, that the classifier assigns the candidate on the tag side with a high probability to the class of B mesons (instead of $\overline{B}$ mesons). These classifier bins are mapped in 7 bins r_i to calculate the corresponding wrong tag fractions with the limits

$$\text{limits}(r_i) = 0.0, 0.1, 0.25, 0.5, 0.625, 0.75, 0.875, 1.0 \tag{5.30}$$

to calculate a metric for the determination of the flavor tagging efficiency. To take into account that different bins have an unequal population, the wrong tag fractions are weighted according to the number of events in the corresponding bins N_i with the efficiency ϵ_i

$$\epsilon_i = \frac{N_i}{\sum_j N_j}. \tag{5.31}$$

Combined, the effective tagging efficiency $\mathcal{Q}$ is

$$\mathcal{Q} = \sum_{i=1}^{7} \epsilon_i (1 - 2w_i)^2. \tag{5.32}$$

The shape of the flavor tagger output $\mathcal{O}$ on Monte Carlo and data is shown in Figure 5.9 for the $B^0 \to D^{*-}\pi^+$ channel and in Figure 5.10 for the $B^- \to D^0\pi^-$ channel. The shapes for the category based approach can be found in Appendix A. The determination of the wrong tag fraction, except for the estimation of the peaking background component does not rely on the difference in normalization between data and MC.

Propagation of Uncertainties

For a proper estimation of the uncertainties of the wrong tag fractions w_i and the effective tagging efficiency $\mathcal{Q}$, the uncertainties of the measured yields in the corresponding flavor tagging bins have to be propagated. The propagation of the uncertainty of a function f from the basis x to the basis y is approximated via a Taylor series, taking only the first order into account. Here the Jacobian J with its components

$$J_{ij} = \frac{\partial y_i}{\partial x_j} \tag{5.33}$$

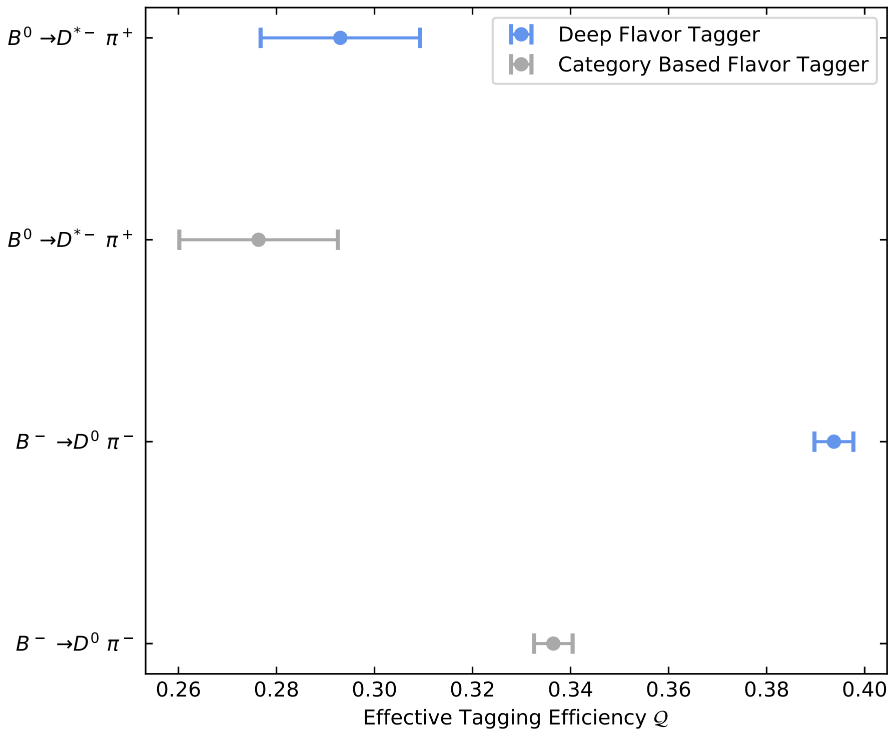

Figure 5.11.: Effective tagging efficiencies for the Deep Flavor Tagger and Category Based Flavor Tagger on channels with neutral and charged B mesons. Note, that the rest of event selection could not be applied to the category based approach in this comparison.

is required. The covariance $C^{(x)}$ of those parameters can then be transformed to the new basis $C^{(y)}$ with

$$C^{(y)} = JC^{(x)}J^T. \tag{5.34}$$

In practice, all uncertainties, using the proper covariance matrices are propagated with the `uncertainties` package [67].

5.5. Final Performance Result

The measurement of the Deep Flavor Tagger is performed on two decay channels. In this section the final performance on data of the flavor tagging algorithm is discussed and compared to the performance of the category based approach. The wrong tag fraction is calculated in the same-flavor-opposite-flavor definition on all channels and an effective tagging efficiency is determined. A similar procedure is applied for the category based approach. Note: Since not the same rest of event selection could be applied to the category based approach, this comparison has to be taken with a grain of salt. Both flavor tagging algorithms operated in a similar regime for neutral B mesons on Belle data.

For the Deep Flavor Tagger an effective tagging efficiency for the charged B meson channel $\mathcal{Q}_{B^+}$ with

$$\mathcal{Q}_{B^+} = 0.3937 \pm 0.0040 \pm 0.0001 \tag{5.35}$$

Table 5.7.: Average wrong tag fractions w and the difference of wrong tag fractions Δw as defined in Section 5.4.1 for the Deep Flavor Tagger on data for the channel $B^0 \to D^{*-}\pi^+$.

	$\bar{w}$	σ_{stat}	σ_{sys}	Δw	σ_{stat}	σ_{sys}
0	0.458525	0.022330	0.000176	-0.032265	0.022330	0.000151
1	0.439102	0.020366	0.000224	0.050380	0.020366	0.000189
2	0.355643	0.018651	0.000507	0.031693	0.018651	0.000117
3	0.268555	0.026686	0.000813	0.005845	0.026686	0.000058
4	0.196921	0.025101	0.001803	0.043595	0.025101	0.001466
5	0.163453	0.024026	0.001183	0.054291	0.024026	0.000213
6	0.028629	0.015714	0.001656	-0.004295	0.015714	0.000119

and for the neutral B meson channel $\mathcal{Q}_{B^0}$ with

$$\mathcal{Q}_{B^0} = 0.2930 \pm 0.0161 \pm 0.0021 \tag{5.36}$$

is measured. These results are visualized in Figure 5.11.

A different performance for the different B meson types is expected. In case of charged B mesons, usually a high momentum charged particle is a very distinct indicator of the B meson flavor. A flavor tagger, which is only trained on neutral B mesons is able to make correct tagging decisions on charged channel. As anticipated, the classifier performs better on charged B meson channels than on the neutral B meson channels on which it is specialized. Nevertheless it provides interesting cross-checks, for consistency of the tagging efficiency and data-Monte Carlo agreement. Due to the different flavor signatures, the weighted averages are calculated for each flavor type separately.

A more detailed evaluation of the wrong tag fractions and the relation of data versus Monte Carlo for the Deep Flavor Tagger is shown in the upper half of Figure 5.12. For a perfect agreement with a perfect simulation, a result on the dashed line within the uncertainties would be obtained. Already small deviations between generated and actual distributions can lead to a shift in tagging performance in different bin ranges. Determining these deviations on control channels is crucial to establish trust in automated methods like the Deep Flavor Tagger and have to be incorporated into analyses which are using this tool. These deviations are not limited to the Deep Learning approach exclusively but have to be investigated for any multivariate method. As a comparison, similar plots are shown for the category based approach in the lower half of Figure 5.12. The wrong tag fractions between data and Monte Carlo are in reasonable agreement for both algorithms on the neutral B meson channels.

For the Deep Flavor Tagger the wrong tag fractions for the channel $B^0 \to D^{*-}\pi^+$ are listed in Table 5.7, the effective tagging efficiencies are listed in Table 5.8. The results on channel $B^- \to D^0\pi^-$ are listed in Table 5.9 and Table 5.10.

For comparison, the effective tagging efficiencies for the charged and neutral channels for the category based approach are shown in Appendix A.

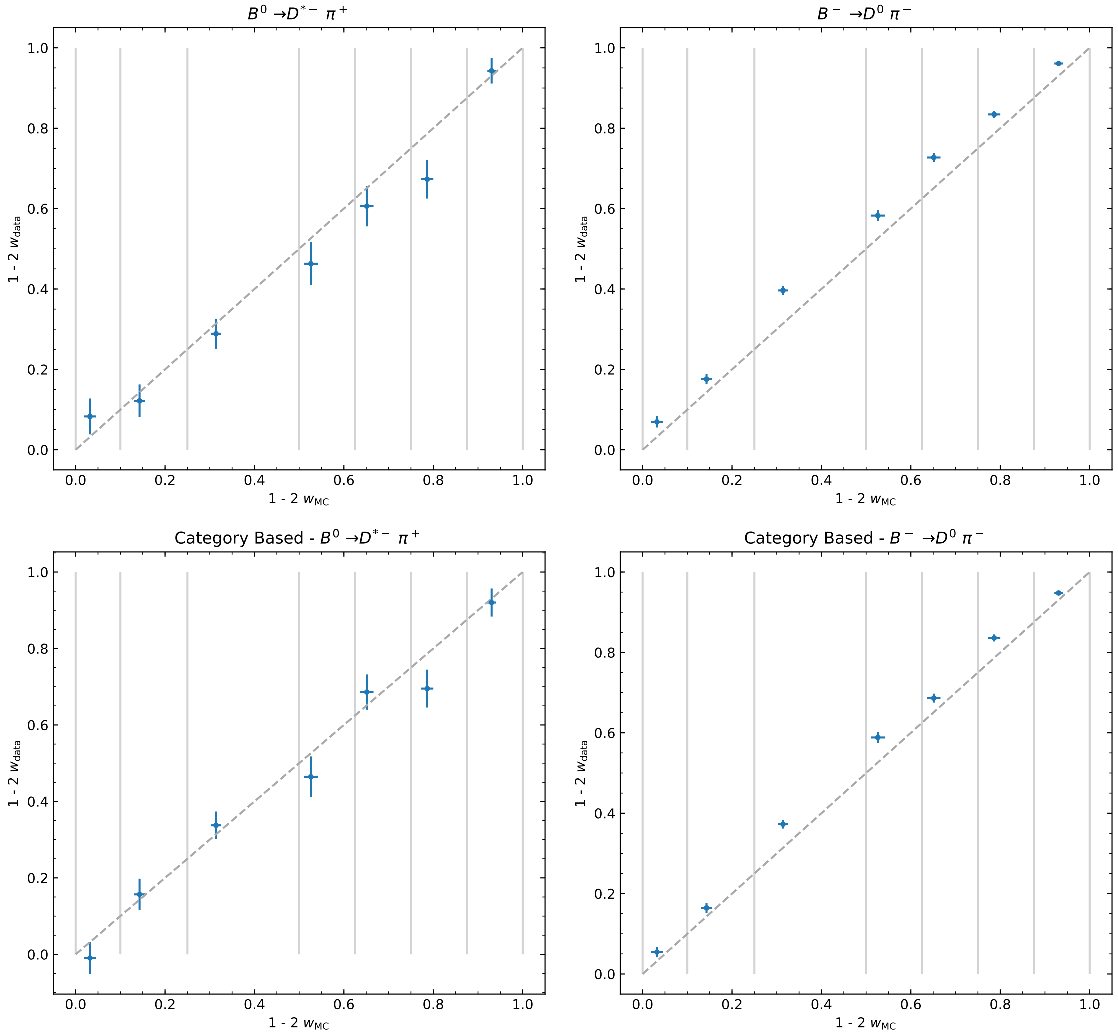

Figure 5.12.: Comparison of the distribution of wrong tag fractions for the deep flavor tagger (upper half) and category based flavor tagger (lower half) between data and Monte Carlo in bins of $r_i = 1 - 2w_i$. A perfect agreement of both distributions, which is not expected due to known differences between data and Monte Carlo, would result in the dashed diagonal line. The solid vertical lines show the the specific classifier bins, defined in Equation (5.30).

Table 5.8.: Average effective tagging efficiencies $\mathcal{Q}$ and the difference of effective tagging efficiencies $\Delta\mathcal{Q}$ as defined in Section 5.4.1 for the Deep Flavor Tagger on data for the channel $B^0 \to D^{*-}\pi^+$.

	$\bar{Q}$	σ_{stat}	σ_{sys}	ΔQ	σ_{stat}	σ_{sys}
0	0.001613	0.001372	0.000013	0.001563	0.001372	0.000013
1	0.004231	0.002192	0.000031	-0.004151	0.002192	0.000031
2	0.017735	0.004473	0.000125	-0.007614	0.004473	0.000054
3	0.019630	0.004585	0.000138	-0.000668	0.004585	0.000012
4	0.037001	0.006055	0.000361	-0.011948	0.006055	0.000264
5	0.046690	0.006730	0.000328	-0.014135	0.006730	0.000101
6	0.166130	0.011707	0.001168	0.002340	0.011707	0.000101
Weighted Sum	0.293030	0.016138	0.002082	-0.034613	0.016138	0.000421

Table 5.9.: Average wrong tag fractions w and the difference of wrong tag fractions Δw as defined in Section 5.4.1 for the Deep Flavor Tagger on data for the channel $B^- \to D^0\pi^-$.

	$\bar{w}$	σ_{stat}	σ_{sys}	Δw	σ_{stat}	σ_{sys}
0	0.465230	0.007050	0.000226	-0.019658	0.007050	0.000226
1	0.412103	0.006411	0.000006	0.014884	0.006411	0.000006
2	0.301786	0.005517	0.000027	0.001247	0.005517	0.000027
3	0.208679	0.006998	0.000022	0.004744	0.006998	0.000022
4	0.136430	0.005626	0.000392	0.017067	0.005626	0.000392
5	0.082840	0.004281	0.000011	-0.001531	0.004281	0.000011
6	0.019511	0.001592	0.000004	0.000973	0.001592	0.000004

Table 5.10.: Average effective tagging efficiencies $\mathcal{Q}$ and the difference of effective tagging efficiencies $\Delta\mathcal{Q}$ as defined in Section 5.4.1 for the Deep Flavor Tagger on data for the channel $B^- \to D^0\pi^-$.

	$\bar{Q}$	σ_{stat}	σ_{sys}	ΔQ	σ_{stat}	σ_{sys}
0	0.000878	0.000306	3.763129e-06	0.000756	0.000306	3.763129e-06
1	0.005090	0.000736	7.174607e-07	-0.001661	0.000736	7.174607e-07
2	0.029584	0.001675	1.013253e-05	-0.000321	0.001675	1.013253e-05
3	0.030828	0.001566	3.964266e-06	-0.001342	0.001566	3.964266e-06
4	0.053203	0.001832	9.090138e-05	-0.004100	0.001832	9.090138e-05
5	0.077927	0.001953	5.284838e-06	-0.001049	0.001953	5.284838e-06
6	0.196219	0.002324	1.522132e-05	-0.002093	0.002324	1.522132e-05
Weighted Sum	0.393729	0.003968	1.048685e-04	-0.009811	0.003968	1.048685e-04

6. Combined Flavor Tagging Performance

In this chapter a combined approach of the category based and the deep neural network based flavor tagging algorithm is discussed. Possible advantages and disadvantages are elaborated on. The following sections focus on leveraging possible correlations between features of the deep neural network flavor tagger and the tagging decision of the category based approach.

6.1. Algorithm Attributes

In Chapter 4 it was indicated that the predictions of the category based and the deep neural network approach are intrinsically different. For instance, both classifiers have a different shape (see Figure 4.6). Besides that, the representation of data is one of the most important aspects in classification. Both features of the algorithms are constructed very differently. This is elaborated on in the following.

The category based approach uses a representation, which is generated by sub-classifiers. These sub-classifiers are trained on different physical constructs, e.g. if a track belongs to a distinct lepton category. At the time of writing, the category based tagger has up to 13 channels. Each of those channels has a specific set of input variables and aims to describe a certain physical aspect. Here expert knowledge and tweaking is used to design these categories. The categories are discussed in detail in Section 3.3.1. The information about correlations of those channels is compressed to a single classifier variable with the help of gradient boost decision trees to infer the charge of the tag-side B meson.

In contrast, the deep neural network based approach generates this representation automatically. The t-SNE distribution of events, shown in Chapter 4, indicate that the categories of this generated representation might not coincide directly with the representation of the category based approach. Nevertheless, the results of Chapter 5 have shown that these representations developed by the algorithm are no artificial fluctuation on Monte Carlo, but are beneficial for the classification on data.

Assuming that none of these representations are perfect, attempts can be made to combine both approaches of flavor tagging for the benefit of a better representation. At the current state, the category based approach includes additional of information e.g. ECL clusters and KLM clusters. These variable types still remain to be implemented for the Deep Flavor Tagger approach. This can contribute to a potential performance improvement.

6.2. Algorithm Combinations

There are multiple ways to combine the information of both algorithms. From the machine learning perspective, the most straightforward way is to exploit the engineered features of the category based algorithm as high level features. It has been shown recently, e.g. [68], that adding high level features can significantly improve the exploitation of correlations with the low level features, and thus also the classifier inference. For this comparison, the output of the category based flavor tagger is used as a high level feature of the deep neural network algorithm during classifier training.

Another suggestion for further studies would be to use the sub-categories from the category based approach as additional features for each track. The information can be passed along with the currently available data set for the deep neural network approach.

6.3. Data Sets and Classifier Preparation

The training data set for the Combined Deep Flavor Tagger was generated similarly to the previous approach, but enriched with the output of the category based flavor tagger. Here, the signal-side B meson decays to two neutrinos with $B^0 \rightarrow \nu\bar{\nu}$, as well. The tag-side B meson decays include the full spectrum of decays defined in the official Belle decay-file. This covers a vast amount of the most relevant B meson decays (see Section 5.2). For the category based Flavor Tagger, the officially provided weight files are used. The validation of the trained classifiers is performed on independent test sets.

6.4. Performance of the Combined Tagger

The performance of the presented algorithm, listed in Table 6.1, shows promising results on Monte Carlo data. On the $B^0 \rightarrow \nu\bar{\nu}$ Monte Carlo data, the combination of both algorithms clearly performs best.

Tests on the signal specific decay channels on $B^0 \rightarrow D^{*-}\pi^+$ and $B^- \rightarrow D^0\pi^-$ show a different picture. The corresponding reconstruction procedure is described in Chapter 5. The results of this study are listed in Table 6.2. It is clearly visible that a combination of both algorithms does not lead to any visible improvement on these channels. Note, that these results may alter with a different rest of event selection. The calibration and validation process is repeated similarly on data to the procedure shown in the previous chapter. Statistical and systematic uncertainties are determined in the same way. In Figure 6.1 these results are visualized. Also here, no improvement using a combination in the described way is visible.

Classifier Name	Channel	Effective Tagging Efficiency $\mathcal{Q}$
Deep Flavor Tagger	$B^0 \to \nu\bar{\nu}$	0.3455 ± 0.0024
Category Based Flavor Tagger	$B^0 \to \nu\bar{\nu}$	0.3419 ± 0.0024
DNN Combination	$B^0 \to \nu\bar{\nu}$	0.3591 ± 0.0024

Table 6.1.: Performance for multiple flavor tagging algorithms on mono-generic Monte Carlo data. Here, the combined tagger here shows clearly a better tagging performance. Due to the clean signature of mono-generic decays, classifiers have a better performance in general.

Classifier Name	Channel	Effective Tagging Efficiency $\mathcal{Q}$
Deep Flavor Tagger	$B^0 \to D^{*-}\pi^+$	0.3306 ± 0.0055
Category Based Flavor Tagger	$B^0 \to D^{*-}\pi^+$	0.3125 ± 0.0054
DNN Combination	$B^0 \to D^{*-}\pi^+$	0.3306 ± 0.0055
Deep Flavor Tagger	$B^- \to D^0\pi^-$	0.4130 ± 0.0011
Category Based Flavor Tagger	$B^- \to D^0\pi^-$	0.3422 ± 0.0012
DNN Combination	$B^- \to D^0\pi^-$	0.3909 ± 0.0012

Table 6.2.: Performance of the Combination of Flavor Taggers on Monte Carlo data for the different decay channels on all streams. Here, no benefits of the combined tagger are recognizable.

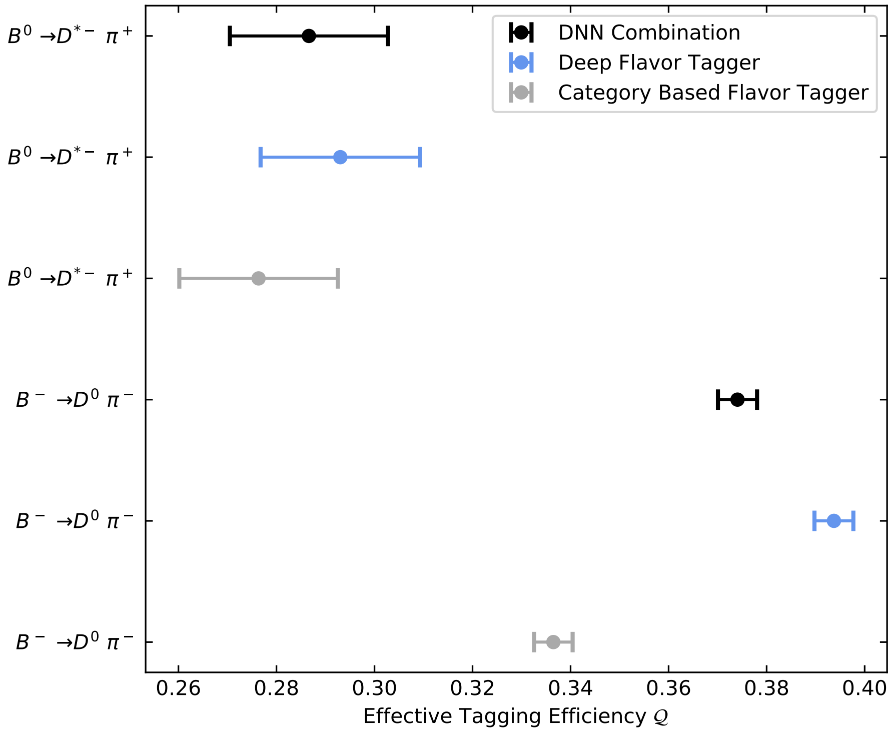

Figure 6.1.: Comparison on data of the different tagging algorithms on data for the $B^0 \to D^{*-}\pi^+$ and $B^- \to D^0\pi^-$ channels. The combination of both algorithms does not lead to an improvement for the tagging performance on data.

The results have shown that a very simplistic combination of both algorithms has short comings. In further studies, it would be worthwhile to study if another combination of the components could lead to improvements of the classification results. A possible option is the usage of all category based sub-categories as low level features for the Deep Flavor Tagger.

Another option is a more signal specific algorithm training. Although previous Monte Carlo studies have shown that a training on mono-generic Monte Carlo is favorable [38], here a signal specific training might have a measurable impact. Furthermore, a more data-driven approach is possible for the Deep Flavor Tagging algorithm since there is no pre-knowledge about physical categories in the data required. It is possible that domain adaption to the data domain can improve the flavor tagging accuracy on independent channels.

7. Performance on Belle II MC and early Belle II Data

In the previous chapters, it has been shown on Belle Monte Carlo data and experimental data, that the Deep Neural Flavor Tagger is a promising approach. Its performance is competitive to the category based approach. Data sets for early runs of the Belle II experiment are available, thus first checks on these data sets can be made. In this chapter, the performance of the Deep Flavor Tagger and the most recent version of the category based flavor tagger are compared on Phase II data of the Belle II experiment.

In an independent study of the flavor tagging performance on Belle II data and MC, F. Abudinén et al. analyzed and compared the category based approach and the DNN based approach with the weights provided in this thesis [69]. This analysis will be summarized in this chapter.

7.1. The Belle II Data Set

The Belle II experiment has conducted multiple Monte Carlo production campaigns, each with either improvements or adjustments in reconstruction algorithms, beam conditions and other means, to achieve a better agreement with data. At the time of writing this thesis, there are several ongoing studies to improve particle identification, impact parameter resolutions and calorimeter reconstruction. Part of this constant improvement process is the validation of the flavor tagging algorithms on the currently available Monte Carlo data as well as real data.

In these studies, the Monte Carlo campaign `MC12b` is used. All good runs of experiment 7 and 8 with an integrated luminosity of 5.15 fb^{-1} of data are selected.

7.2. Performance of the DNN Classifier on Monte Carlo

The classifiers of the category based and DNN based algorithms are trained on mono-generic MC with $B^0 \to \nu\bar{\nu}$ decays. The studies of F. Abudinén et al. on

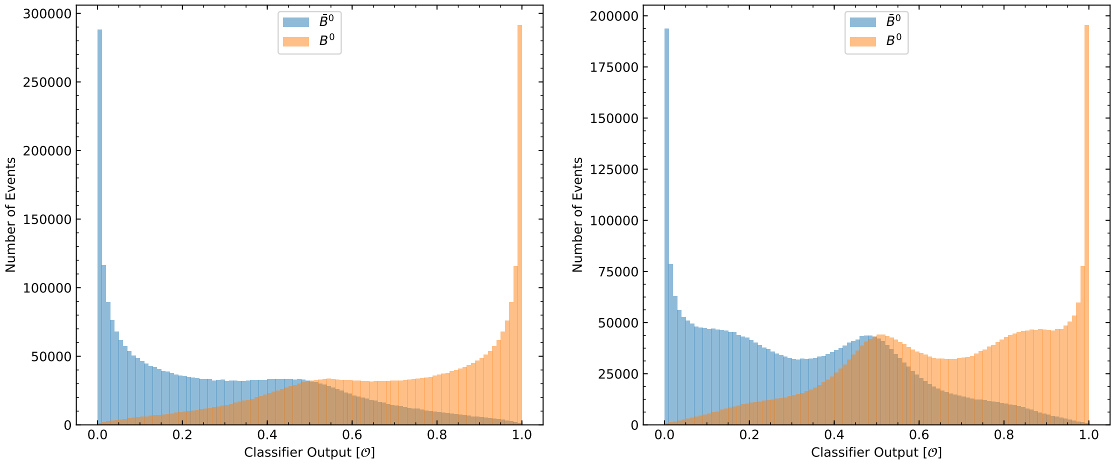

Figure 7.1.: Trained DNN classifiers on `MC12b` Monte Carlo and evaluated on 6 million events. The output of the unmodified classifier is shown on the left hand side, the degraded classifier without impact parameters on the right hand side. The impact of the variable removal has an influence on the shape.

Classifier Name	Channel	Effective Tagging Efficiency $\mathcal{Q}$
DNN Flavor Tagger	$B^0 \to \nu\bar{\nu}$	0.3979 ± 0.0003
Degraded DNN Flavor Tagger	$B^0 \to \nu\bar{\nu}$	0.3459 ± 0.0003

Table 7.1.: Performance for DNN tagging algorithm on mono-generic Monte Carlo data for the `MC12b` campaign with and without restrictions. A definition of the metric, the effective tagging efficiency $\mathcal{Q}$ is described in detail in Section 5.4.

Belle II data show that the impact parameters have large discrepancies between data and MC. At the time of writing, this issue is still under investigation.

To address this, two different types of the flavor tagging algorithm are trained; one with impact parameters as features and a degraded one without. The following variables are dropped in the degraded case:

- the distance from the perigee to the interaction point d_0,
- the distance in z direction from the interaction point z_0,
- the number of hits in the CDC,
- and the p-value of the track fit of the CDC.

With excluded variables such as the distance to the impact point, tagging decisions for data and Monte Carlo are in much better agreement. The shape of both classifiers – with and without the parameter restrictions – are shown in Figure 7.1 on an independent MC test data set.

It is clearly visible that the effective tagging efficiency suffers from the loss of separation power when neglecting those important variables. A comparison for the DNN Flavor Tagger is listed in Table 7.1. Nevertheless, it is expected that the issue is solved in the near future and the variables can be added again.

7.3. Reconstruction Channels for Belle II Data

The reconstruction of the channel is performed by [69]. Due to limited statistics, the B mesons are reconstructed in several decay modes. These are listed below, including charge conjugated decays:

- $D^0 \to K^-\pi^+$
- $D^0 \to K^-\pi^+\pi^0$
- $D^0 \to K^-\pi^+\pi^-\pi^+$
- $D^0 \to K_S^0\pi^+\pi^-$
- $D^+ \to K^-\pi^+\pi^+$
- $D^+ \to K_S^0\pi^+$
- $D^{*-} \to D^0\pi^+$
- $a_1^+ \to \pi^+\pi^+\pi^-$
- $\rho^+ \to \pi^+\pi^-$.

These modes are reconstructed to B mesons in the following channels:

- $B^0 \to D^{*-}\pi^+$
- $B^0 \to D^{*-}\rho^+$
- $B^0 \to D^{*-}a_1^+$
- $B^0 \to D^-\pi^+$
- $B^0 \to D^-\rho^+$.

The shape of the classifier is extracted from data with the so-called ${}_s\mathcal{P}lot$ technique [70]. Here, a maximum likelihood fit is performed for the discriminating variable and an independent control variable. The procedure allows to unfold two distributions directly on data. Using an independent discriminating variable allows the distributions to be independent of Monte Carlo assumptions.

In this case, as control variable, the flavor tagger output $q \cdot r$ is used, which corresponds to the flavor tagger output of the previous sections $\mathcal{O}_{\mathrm{FT}}$ with

$$q \cdot r = 2\mathcal{O}_{\mathrm{FT}} - 1. \tag{7.1}$$

For the discriminating variable the beam constrained mass M_{bc} is selected.

Signal and background regions with clear distinct shapes are defined. Events in the signal region are weighted with so-called ${}_sWeights$, according to the expected signal in this region. ${}_sWeights$ can also be negative. The following distinct shapes are considered:

- The signal shape, modeled as two Gaussian functions
- and the continuum background, modeled with an ARGUS function.

The fraction of combinatorial background is determined on Monte Carlo and coupled to the signal yield. This allows to reconstruct the classifier shapes for continuum and neutral B mesons decays.

7.4. Classifier Shape Comparison on Belle II Data

The classifier shape of the reduced variable set for the Deep Flavor Tagger and the category based flavor tagging algorithm is shown in Figure 7.2. In the upper panel, the signal yield on M_{bc} is used as a discriminating component. Furthermore, an additional cut on $M_{\text{bc}} > 5.27$ GeV has been performed to get a cleaner sample. The measured yield for the binned classifier output are in good agreement between data and Monte Carlo.

As a comparison, in the lower half of Figure 7.2, the shape of the classifier for the continuum component at energies of the $\Upsilon(4S)$ resonance is shown. Since there are no B mesons expected in this component, a different classifier shape is expected. The signatures of these events differ significantly from B meson decays, with manifests itself in a more centralized distribution.

Although both classifiers suffer from a reduced variable set due to the absence of the impact parameters, they perform reasonably well in tests with data and MC. These results show that both flavor tagging algorithms are ready to be used (and to be calibrated) on Belle II data.

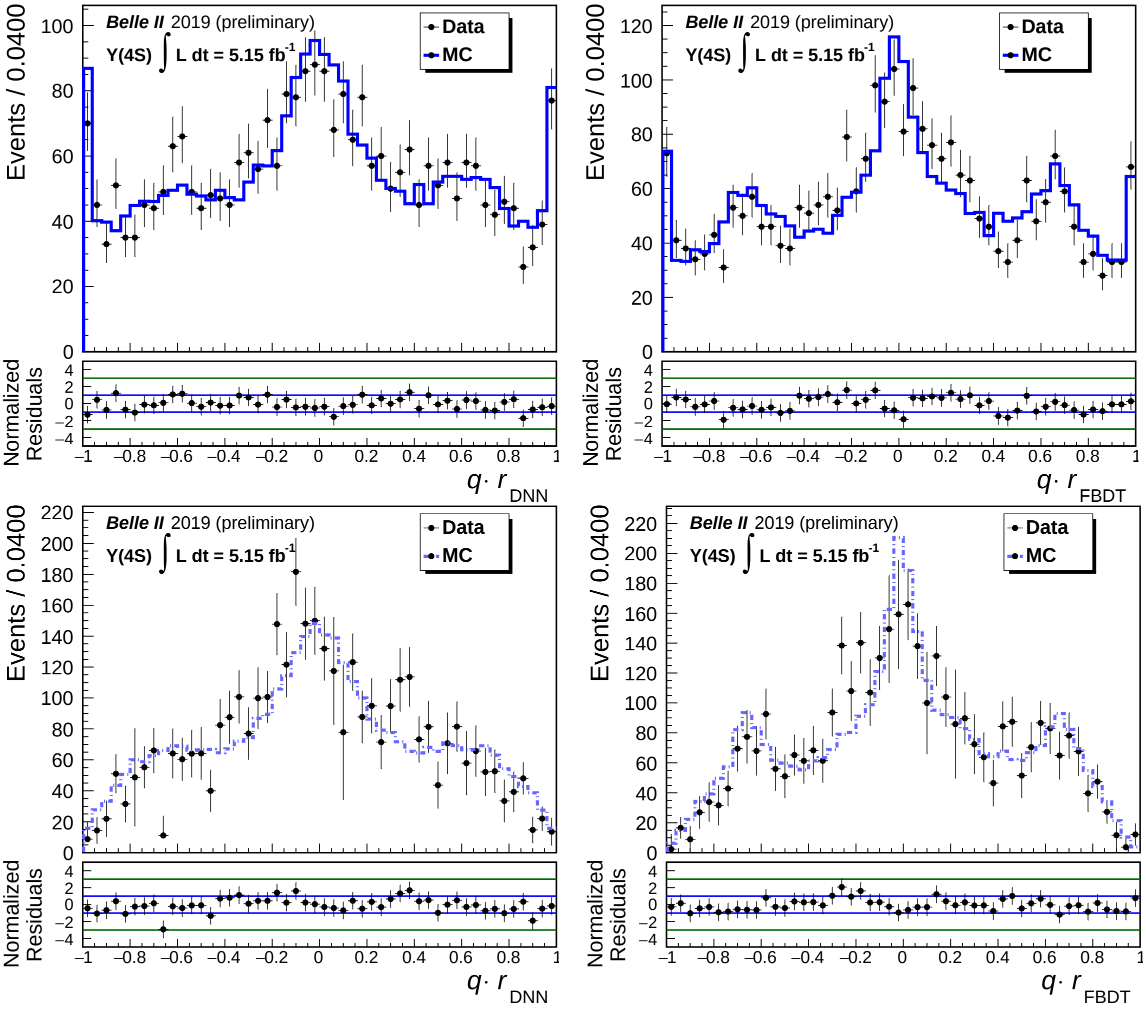

Figure 7.2.: Classifier shapes for the DNN flavor tagging (left hand side) algorithm and the category based algorithm (right hand side) on data and Monte Carlo. The shapes are shown for $\Upsilon(4S)$ events for fitted signal events in the signal window with $m_{\mathrm{bc}} > 5.27$ GeV (upper half) and continuum events (lower half). Taken from [69].

8. Machine Learning at Belle II with Wasserstein Generative Adversarial Networks

Detector responses of particle decays are known to be simulated with high accuracy [71]. In particular this is the case for particle showers in calorimeters. Still, these simulations require a large fraction of computational resources of the overall simulation process. In case of an electromagnetic calorimeter, stochastic processes determine the description of production and interaction of particle showers in the detector material. Each component has to be simulated by individual steps.

Machine learning methods provide the opportunity to improve the accuracy and precision in their fields of application or instead to speed up the runtime by several orders of magnitude [72]. In this chapter, a proof of concept study for the simulation of the energy deposition in the electromagnetic calorimeter at Belle II is presented. It is studied, if the spatial energy distribution of particle cascades can be obtained with adequate accuracy. The results in this chapter are derived from work done in collaboration with Jubna Irakkathil Jabbar.

8.1. WGAN for Future Fast Simulations at Belle II

Machine learning has been applied very successfully to image classification and generation. Starting with the classification of handwritten letters with convolutional neural networks [73], image classification has been improving continually. These techniques are not only used for discrimination, but also for the creation of new objects with specific properties, referred to as generative algorithms.

Generative adversarial networks (GANs) [74] are designed to to generate a distribution P_{gen} from data P_{data} in an unsupervised manner. Here a generator G and a discriminator D compete against each other in a minimax non-cooperative game. The discriminator network aims to distinguish generated objects from real objects. In contrast, the generator aims to maximize the probability that the discriminator network makes a mistake in discriminating. During the training procedure, a common loss function $\mathcal{L}(D, G)$ is optimized with

$$\min_{G} \max_{D} \mathcal{L}(D, G) = \mathbb{E}_{P_{\text{data}}}[\log D(\mathbf{x})] + \mathbb{E}_{P_{\text{gen}}}[\log(1 - D(G(\mathbf{z})))] \tag{8.1}$$

where $\mathbf{x}$ represents the data, and $\mathbf{z}$ the input to the generator, which can be chosen arbitrarily. Here $\mathbb{E}_{P_{\text{data}}}$ and $\mathbb{E}_{P_{\text{gen}}}$ denote the expectation value according to the density on data and on generated samples, respectively.

The weights of the generator θ_G and the weights of the discriminator θ_D can be optimized by determining the gradients with respect to the individual parts of the loss function. In principle, the discriminator can always be minimized for a given generator, if both models are chosen correctly. While the training procedure is fairly robust, in practice GANs have also some disadvantages. The representation of the generator changes during the training procedure and the discriminator has to be adapted. Therefore, usually both parts are trained iteratively on mini batches – the training of generator and discriminator has to be carefully balanced to prevent the situation that too many different values of z are mapped to a similar solution. This situation is called *mode collapse* and represents one of the major challenges during the training procedure of GANs.

There are many different metrics to compare the distributions of the generator P_{gen} and training data P_{data}. For instance the Kullback-Leibler divergence, see Equation (4.11), or a symmetrized combination can be used. A more stable solution for the training procedure can be constructed with the so-called Wasserstein Generative Adversarial Neural Networks (WGANs) [75]. As a distance metric W the Wasserstein-1 distance is used. It can be constructed by using the Kantorovich-Rubinstein duality with

$$W(P_{\text{data}}, P_{\text{gen}}) = \sup_{\|f\|_L \leq 1} \mathbb{E}_{P_{\text{data}}}[f(\mathbf{x}; \theta_D)] - \mathbb{E}_{P_{\text{gen}}}[f(\mathbf{x}; \theta_{\mathbf{G}})]. \tag{8.2}$$

Here, f denotes a 1-Lipschitz function with $f : [0, 1] \to \mathbb{R}$ which obeys

$$|f(x_1) - f(x_2)| \leq |x_1 - x_2| \qquad \forall x_1, x_2 \in [0, 1]. \tag{8.3}$$

Note that the interval $[0, 1]$ marks the output space of the individual metrics. Since the discriminative part is not trained directly to classify, it is often referred to as the critic. In practice, for finding the supremum of f which satisfies the 1-Lipschitz condition, a neural network is used. The initial version uses weight clippings of the neural network that resembles f after each iteration step. The advantages of a WGAN manifest during the training procedure in the gradients of the critic, which are more evenly distributed.

To stabilize the training procedure and obtaining a better value surface of the critic, a regularization penalty $\mathcal{L}_\lambda$ can be added to the loss function [76] instead of weight clipping. The samples $\hat{\mathbf{x}}$ are chosen from an uniform distribution between the generated and the real data points. This parametrization defines the distribution of the penalty term $P_{\hat{\mathbf{x}}}$. With this definition, the 1-Lipschitz condition is enforced with

$$\mathcal{L}_\lambda = \lambda \; \mathbb{E}_{P_{\hat{\mathbf{x}}}}[(||\nabla_{\hat{\mathbf{x}}} D(\hat{\mathbf{x}})||_2 - 1)^2] \tag{8.4}$$

where the strength of the regularization can be adjusted with the hyper-parameter λ and $||.||_2$ denotes the Euclidean norm.

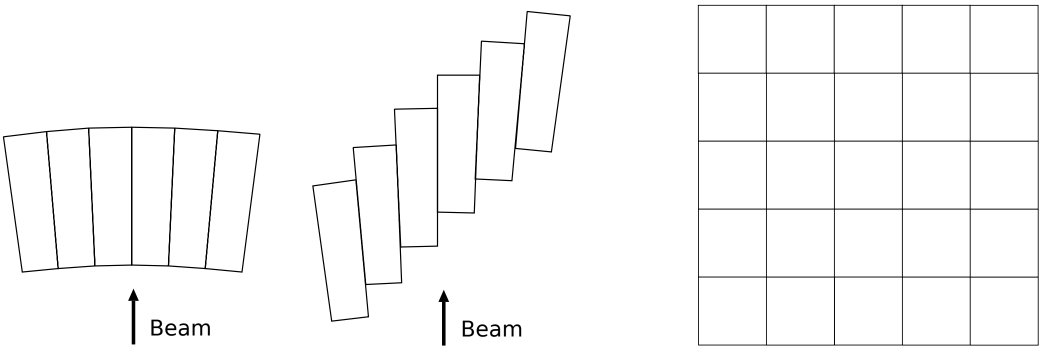

Figure 8.1.: Schematics of possible configurations of the CsI(Tl) crystals of the ECL, adapted from [1]. Schematics of the simplified configuration used for the proof-of-concept study is shown on the right hand side.

In high energy particle physics, there have been several advances to replace computationally expensive simulations with deep generative models. At present, generating calorimetric images still requires a relatively large amount of computing power compared to other steps; resources which can be used elsewhere. Calorimetry imagery is one area where the recorded data resembles the established data sets of image recognition, allowing the exploitation of these techniques for physics. For instance in [72], the generation of images of the detector response for particle jets with a GAN is investigated. Reductions of the computing time of several orders of magnitudes are possible. In [77], the detector response for a high granular calorimeter with multiple layers at the CMS experiment at the Large Hadron Collider is investigated.

As part of this thesis, we took the first steps to produce calorimetric images at Belle II using Wasserstein Generative Adversarial Neural Networks, based on the approach of [77].

8.2. A Simple Model for the Belle II Electromagnetic Calorimeter

In Section 2.1.2, the detector parts of the Belle experiment are described. For Belle II, the crystals of the Belle electromagnetic calorimeter (ECL) are mostly reused. For more details about the ECL, see [1]. A schematic of the complete calorimeter is shown in Figure 2.4. The crystal configurations are different in the barrel and the end-cap part. On the left hand side of Figure 8.1 different crystal alignments are shown.

Charged and neutral particles cause photon electron showers, which propagate in a cascade. The deposited energy is measured with photo-diodes from the intensity of the scintillation light. The crystal size was chosen so, that roughly 80% of the energy of a photon is deposited in a single crystal. The front sizes vary from 44.5 mm to 70.8 mm; typical front size is 55 mm × 55 mm. The rear size differs from the front size. For the proof-of-concept simulation study, a more simple setup of an array with 5×5 crystals with an 60 mm × 60 mm × 300 mm was chosen. Like the ECL, the base material of CsI(Tl) was selected for the simulation of the crystals. The data sets are generated with `Geant4` with an energy range from 0.5 GeV to 3.0 GeV in steps of 0.5 GeV. The distribution of the generated data is visualized in Figure 8.2.

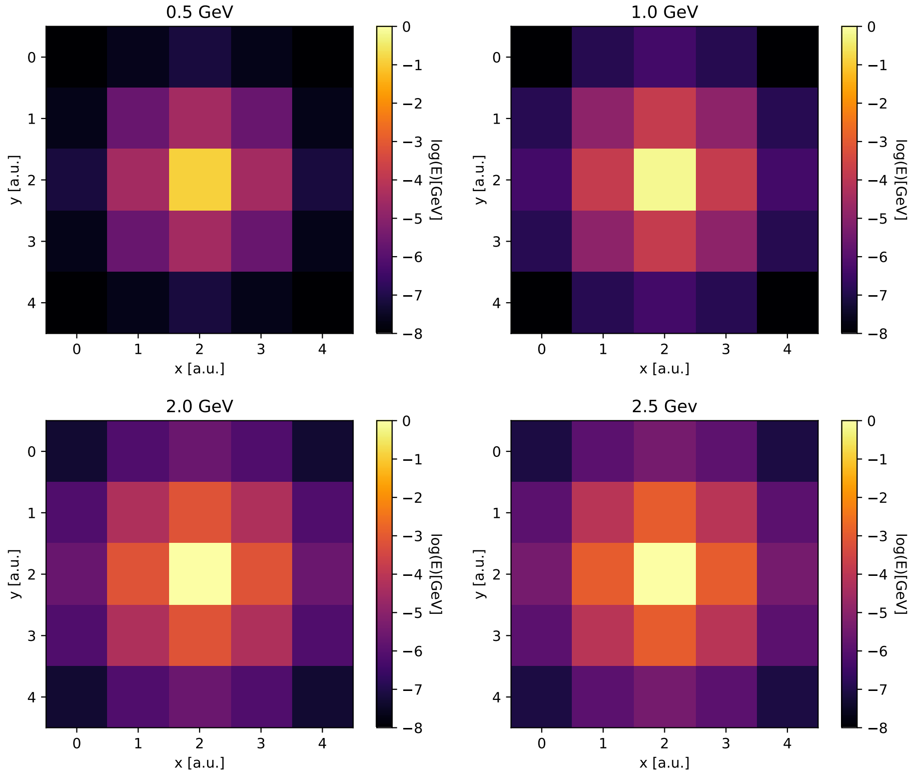

Figure 8.2.: Energy response of the toy calorimeter to an electron with energies of 0.5 GeV, 1.0 GeV, 2.0 GeV and 2.5 GeV. Shown are the averages over the samples of the Monte Carlo data set generated with `Geant4`. The x and y labels mark the crystals. The energy deposited in each crystal is displayed on a log scale.

For the test setup the interaction of electrons with different energies with detector was simulated with `GEANT4`. The obtained samples, except the 2.0 GeV and 3.0 GeV samples, were used as training data for the WGAN architecture.

8.3. Conceptual Study for ECL Fast Simulation

For the training of the WGAN model an architecture based on [77] is chosen. Here, an additional approach for label conditioning is applied, which was proposed by [78] since deep generative models tend to neglect important attributes of physical quantities during data creation. Here, an additional network z_i is introduced to evaluate if the produced samples reflect a specific physical quantity accordingly. In case of the ECL simulation, the total energy of a produced event is chosen as physical quantity. The weights of the label condition network are trained constantly during the critic training in a supervised manner by using a mean squared error $\mathcal{L}^c$

$$\mathcal{L}_i^{\mathrm{c}} = (y_i - z_i(\mathbf{x}))^2 \tag{8.5}$$

with the truth labels y_i. Since the attributes of the constrainer network are continuously changing during the training procedure, the generator loss is extended by the difference $\mathcal{L}^{\mathrm{c,aux}}$ of the conditioning loss on the MC truth $\mathcal{L}^{\mathrm{truth}}$ samples and by the WGAN architecture trained sample $\mathcal{L}^{\mathrm{gen}}$. The auxillary loss then is given by

$$\mathcal{L}^{\mathrm{c,aux}} = \kappa \sum_i |\mathcal{L}_i^{\mathrm{c,truth}} - \mathcal{L}_i^{\mathrm{c,gen}}|, \tag{8.6}$$

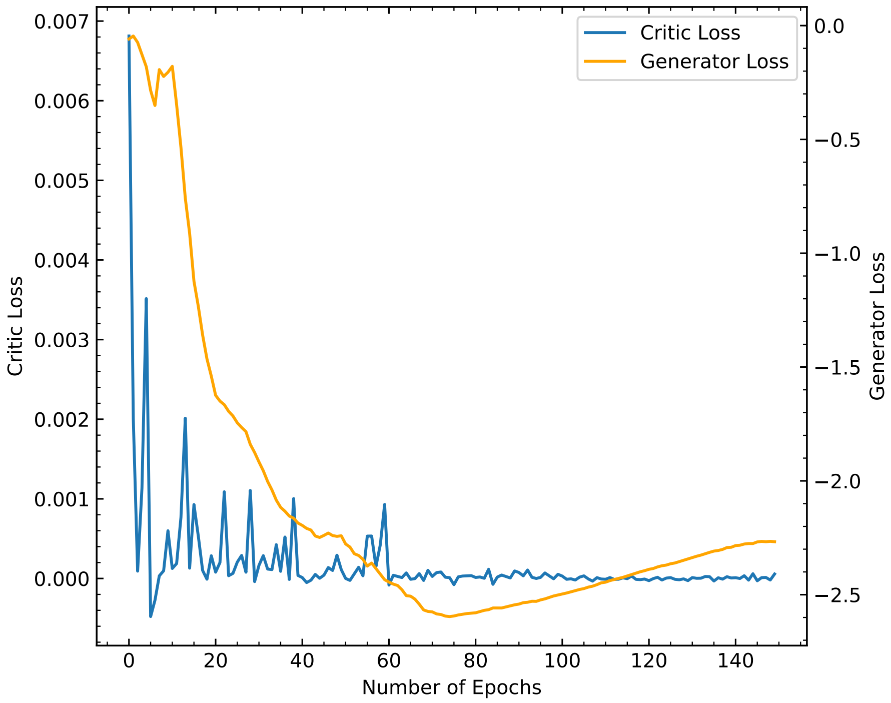

Figure 8.3.: Loss functions of the generator and the critic during the training procedure. Note that the label conditioning term $\mathcal{L}^{\mathrm{c,aux}}$ is not contained in this illustration.

where κ denotes a hyper-parameter for the impact of the auxillary term. For this study, as constraining quantity the total energy deposited in the area of the 5×5 crystals is chosen. The loss function during the training procedure is shown in Figure 8.3. One can see that the generator loss increases at several epochs of the training. This is not a surprise, since the critic also improves separability between samples. Nevertheless it is clearly visible, that the losses and the generated distributions improve over the training steps. To take into account energy thresholds of the ECL, energy depositions below a value of 10^{-3} GeV are neglected. An average of generated images at different states of the training procedure is shown in Figure 8.4.

As an additional remark, based on [77], the WGAN architecture is also able to interpolate between energy distributions. Even thought the algorithm has not seen samples with an energy of 2.0 GeV during the training procedure, it is still able to generate images at this energy. A comparison of the total energy and the energy of an active pixel between true and generated images is shown in Figure 8.5. Still, there are discrepancies between both distributions visible and are subject to current research.

Since the overall performance is very promising, further work is invested to expanded this approach for different geometries of the actual ECL calorimeter, shown in the left hand side of Figure 8.1.

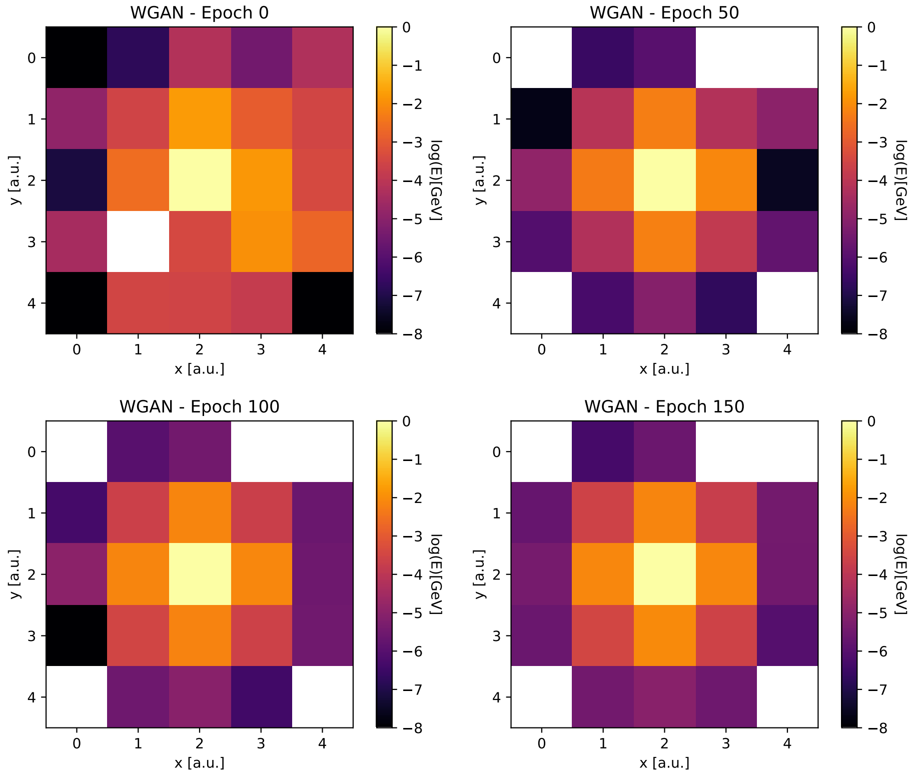

Figure 8.4.: Energy response of the toy calorimeter to an electron with an energy of 2.5 GeV. Shown are the averages over 2500 samples for the WGAN architecture during algorithm training for different epochs. The x and y labels mark the crystals. The energy deposited in each crystal is displayed on a log scale.

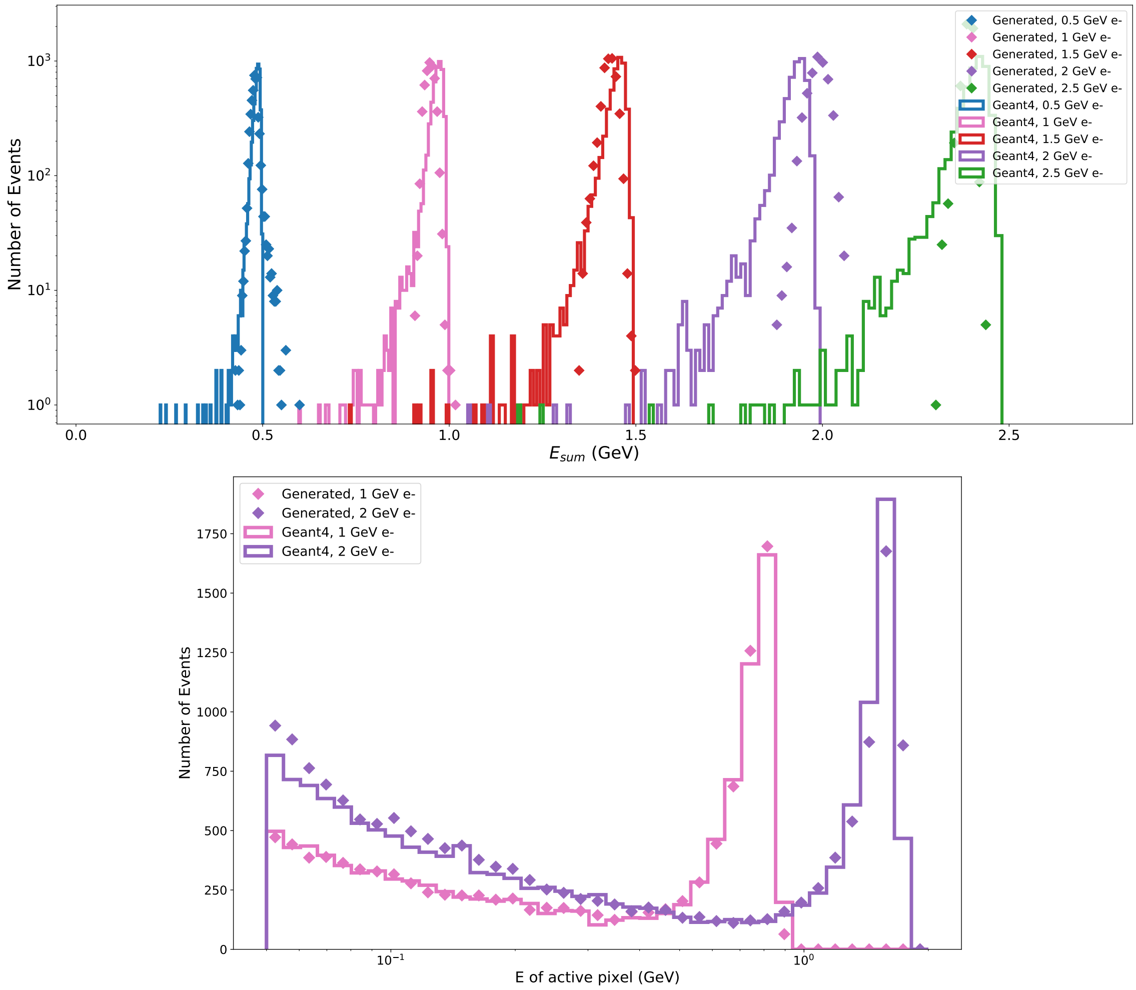

Figure 8.5.: Comparison between true images and images generated by a fully trained WGAN. Each color represents an energy of the electron. In the upper half, the sum of all energy deposition is compared. In the lower half, the energy of the active pixels is compared. No samples with 2.0 GeV were shown to the WGAN during the training procedure.

9. Summary and Outlook

In this thesis a novel algorithm for neutral B meson flavor tagging based on deep neural networks for the Belle II experiment is developed, and calibrated on the neutral B meson channel $B^0 \to D^{*-}\pi^+$ and on the charged B meson channel $B^- \to D^0\pi^-$. This algorithm is the first deep learning based machine learning method at Belle and Belle II, opening uncharted territories for improvements in physics analyses. The calibration is performed on the full $\Upsilon(4S)$ data set of the Belle experiment with 772 million B meson pairs. For the conversion of Belle data and the reconstruction of the decays, the Belle II analysis software is used.

For the analysis procedure, the signal side B meson is reconstructed in one of the two aforementioned decay channels. The tag side B meson is not reconstructed explicitly, instead unique signatures of the event are used, to infer the B meson flavor from its decay products. This so-called flavor tagging is performed with a deep neural network algorithm that develops its own representation of the data during the training process. Since potential difference between Monte Carlo data and recorded data could have a significant influence on the algorithm, a careful validation on data was performed.

Background events are reduced in the data samples by using a specific set of cuts and by choosing the best reconstructed candidate for each event. The signal yield is extracted with an unbinned maximum likelihood fit – a kinematic selection is performed on the beam constrained mass M_{bc}. The peaking background components are determined and estimated on Monte Carlo data. The systematic uncertainties are estimated.

The effective tagging efficiency is determined for both channels. Here, the extraction fit is repeated in 14 bins of the classifier output, respectively. The deep neural network based algorithm achieves an effective tagging efficiency on data

$$\mathcal{Q}_{B^+} = 0.3937 \pm 0.0040 \pm 0.0001$$

for the charged B meson channel and

$$\mathcal{Q}_{B^0} = 0.2930 \pm 0.0161 \pm 0.0021$$

for the neutral B meson channel, where the first uncertainty is statistical, the second systematic. A comparison with an alternative, category based flavor tagging algorithm is performed directly on data, see Figure 5.11.

A study of the Belle II readiness of both algorithms shows, that both algorithms are ready to be calibrated on data of the Belle II experiment.

Versatile deep learning methods have been explored in this thesis. First studies of an approach for the usage of deep learning to speed up the simulation process of the electromagnetic calorimeter are presented.

Nevertheless, deep learning methods are still a very young domain at the Belle II experiment. Direct adjustments to the flavor tagging algorithm, for instance adding specific high level features from the category based approach or cluster variables to current configuration are expected to improve results. Domain adaptation on data during the training process has the potential to decrease differences between data and Monte Carlo. This has to be done very carefully to not bias for CP violation quantities, which are intended to be measured. There are many different types of machine learning algorithms which potentially could improve the tagging ability of the deep neural network approach. Promising candidates are self-attention neural networks [79] or graph generative models in combination with reinforcement learning.

Exploring this vast area of potential methods might improve the inference models in such a way that the borders between inclusive methods like the flavor tagging algorithm and exclusive methods like the Full Event Interpretation begin to vanish.

Appendix

A. Measurements for the Category Based Approach

In this section the measurements on data for the channels $B^0 \rightarrow D^{*-}\pi^+$ and $B^- \rightarrow D^0\pi^-$ are listed.

The wrong tag fractions for each bin are shown in Table A.1 and Table A.3. The effective tagging efficiencies for each bin are shown in Table A.2 and Table A.4. The systematic uncertainties for each component for the category based approach are shown in Table A.5 and Table A.6.

The fitted classifier shapes on data for the category based approach visualized in Figure A.1 and Figure A.2.

Table A.1.: Average wrong tag fractions w and the difference of wrong tag fractions Δw as defined in Section 5.4.1 for the Category Based Flavor Tagger on data for the channel $B^0 \rightarrow D^{*-}\pi^+$.

	$\bar{w}$	σ_{stat}	σ_{sys}	Δw	σ_{stat}	σ_{sys}
0	0.504756	0.020936	0.000096	0.028375	0.020936	0.000137
1	0.421582	0.020530	0.000293	-0.030070	0.020530	0.000146
2	0.331252	0.018039	0.000593	0.027692	0.018039	0.000107
3	0.267730	0.026554	0.000818	0.037837	0.026554	0.000155
4	0.156999	0.023032	0.001223	0.018192	0.023032	0.000237
5	0.152423	0.024739	0.001228	0.017250	0.024739	0.000167
6	0.039872	0.018300	0.001614	0.017718	0.018300	0.000101

Table A.2.: Average effective tagging efficiencies $\mathcal{Q}$ and the difference of effective tagging efficiencies $\Delta\mathcal{Q}$ as defined in Section 5.4.1 for the Category Based Flavor Tagger on data for the channel $B^0 \to D^{*-}\pi^+$.

	$\bar{Q}$	σ_{stat}	σ_{sys}	ΔQ	σ_{stat}	σ_{sys}
0	0.000549	0.000796	0.000005	0.000186	0.000796	0.000003
1	0.004942	0.002429	0.000037	0.003332	0.002429	0.000027
2	0.024584	0.005209	0.000173	-0.007529	0.005209	0.000054
3	0.020613	0.004696	0.000145	-0.005990	0.004696	0.000044
4	0.052197	0.007154	0.000377	-0.008228	0.007154	0.000108
5	0.047303	0.006832	0.000335	-0.008264	0.006832	0.000075
6	0.126162	0.010491	0.000885	-0.007022	0.010491	0.000058
Weighted Sum	0.276350	0.016052	0.001937	-0.033516	0.016052	0.000243

Table A.3.: Average wrong tag fractions w and the difference of wrong tag fractions Δw as defined in Section 5.4.1 for the Category Based Flavor Tagger on data for the channel $B^- \to D^0\pi^-$.

	$\bar{w}$	σ_{stat}	σ_{sys}	Δw	σ_{stat}	σ_{sys}
0	0.472691	0.006545	0.000060	0.000474	0.006545	0.000060
1	0.417790	0.006253	0.000027	0.011162	0.006253	0.000027
2	0.313697	0.005412	0.000059	-0.010211	0.005412	0.000059
3	0.205786	0.006845	0.000041	0.007289	0.006845	0.000041
4	0.156832	0.005567	0.000015	-0.005975	0.005567	0.000015
5	0.081980	0.004391	0.000153	-0.007832	0.004391	0.000153
6	0.026065	0.002124	0.000021	-0.003207	0.002124	0.000021

Table A.4.: Average effective tagging efficiencies $\mathcal{Q}$ and the difference of effective tagging efficiencies $\Delta\mathcal{Q}$ as defined in Section 5.4.1 for the Category Based Flavor Tagger on data for the channel $B^- \to D^0\pi^-$.

	$\bar{Q}$	σ_{stat}	σ_{sys}	ΔQ	σ_{stat}	σ_{sys}
0	0.000474	0.000227	0.000002	-0.000013	0.000227	0.000002
1	0.004594	0.000696	0.000003	-0.001204	0.000696	0.000003
2	0.027639	0.001628	0.000023	0.002747	0.001628	0.000023
3	0.032905	0.001613	0.000017	-0.001746	0.001613	0.000017
4	0.054019	0.001911	0.000005	0.002254	0.001911	0.000005
5	0.077055	0.001952	0.000044	0.003987	0.001952	0.000044
6	0.139803	0.002073	0.000035	0.000077	0.002073	0.000035
Weighted Sum	0.336489	0.003930	0.000055	0.006102	0.003930	0.000055

Table A.5.: Systematic uncertainties for the channel $B^0 \to D^{*-}\pi^+$ in relative values for the effective tagging efficiency $\mathcal{Q}$. The systematic uncertainties for the signal shape cancel out of the fractions and will be neglected.

Efficiency Systematics		[%]
	χ_d	0.700637
Signal	Branching Fraction	0.012036
	Number of B Meson Pairs	0.009274
	Particle Identification Correction	0.008779
	Track Reconstruction Efficiency	0.009658
Peaking Background	Branching Fraction	0.020123
	Number of B Meson Pairs	0.008645
	Particle Identification Correction	0.002995
	Track Reconstruction Efficiency	0.003746

Table A.6.: Systematic uncertainties for the channel $B^- \to D^0\pi^-$ in relative values for the effective tagging efficiency $\mathcal{Q}$. The systematic uncertainties for the signal shape cancel out of the fractions and will be neglected.

Efficiency Systematics		[%]
Signal	Branching Fraction	0.001143
	Number of B Meson Pairs	0.001728
	Particle Identification Correction	0.001005
	Track Reconstruction Efficiency	0.001381
Peaking Background	Branching Fraction	0.006650
	Number of B Meson Pairs	0.012492
	Particle Identification Correction	0.005623
	Track Reconstruction Efficiency	0.005772

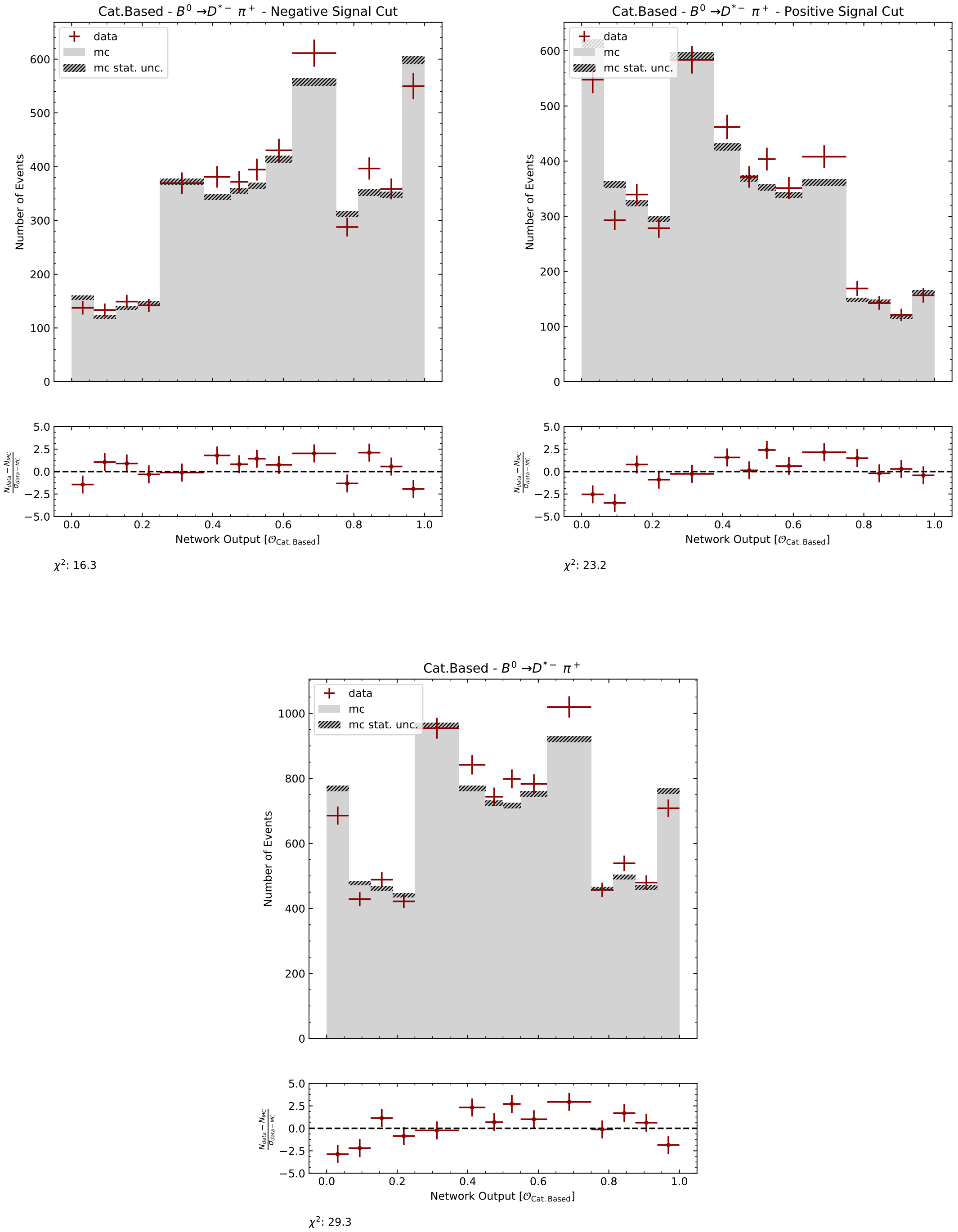

Figure A.1.: Category Based Flavor Tagger shapes of the $B^0 \to D^{*-}\pi^+$ decay, separated by the charge of the reconstructed signal B meson (upper left and upper right). The Monte Carlo shape is shown as a histogram, the fitted yields of the data sample are shown in the corresponding bins. A signal side flavor unspecific view for a mere shape comparison is shown in the lower part.

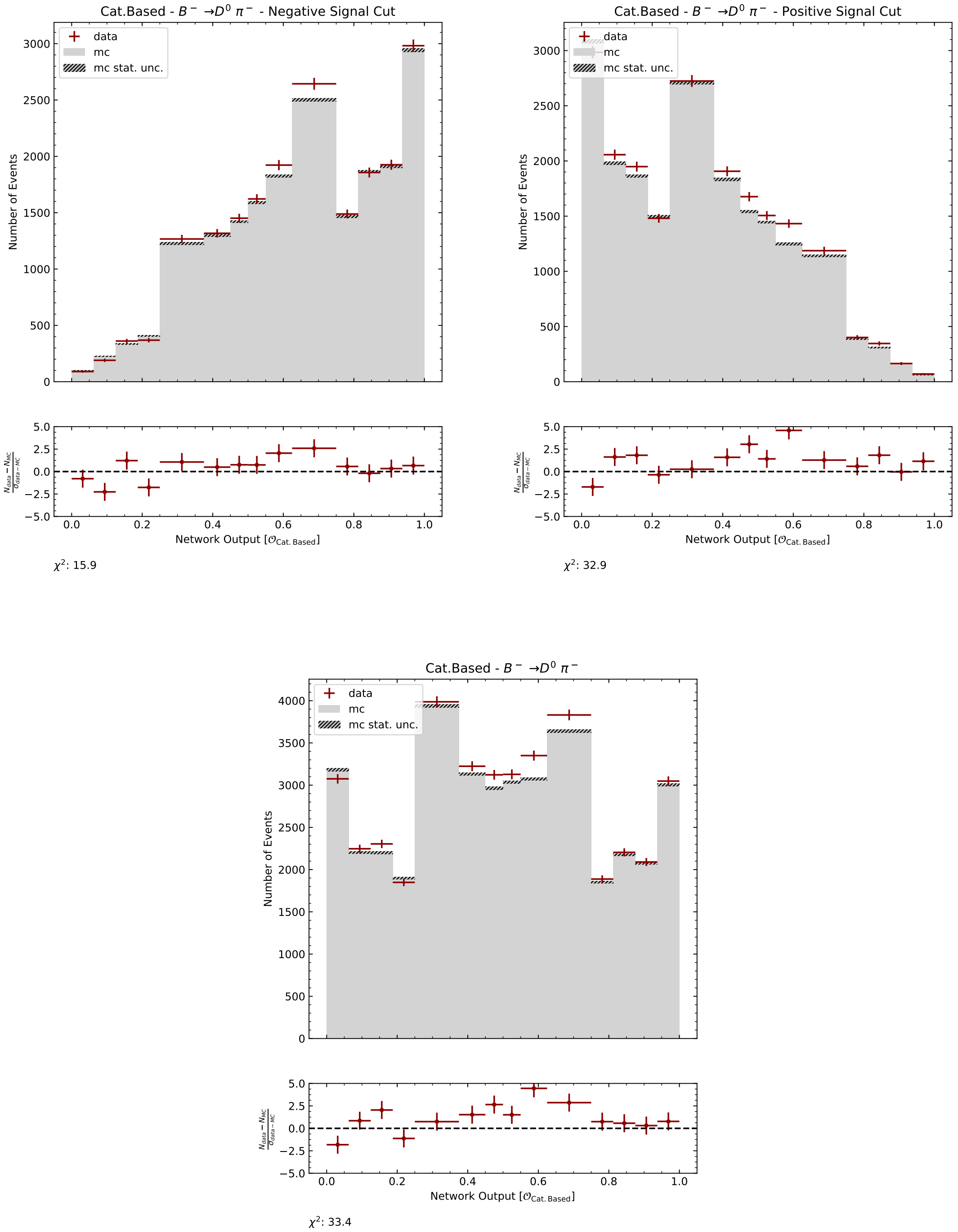

Figure A.2.: Category Based Flavor Tagger shapes of the $B^- \to D^0\pi^-$ decay, separated by the charge of the reconstructed signal B meson (upper left and upper right). The Monte Carlo shape is shown as a histogram, the fitted yields of the data sample are shown in the corresponding bins. A signal side flavor unspecific view for a mere shape comparison is shown in the lower part.

B. Symmetry Bias Studies

In this section, a possible symmetry bias of the classifier is investigated and quantified. Some differences B and $\overline{B}$ meson distributions are expected due to different detector sensitivities for matter and anti-matter particles of the detector. Nevertheless, studies on the category based tagger [80] indicate that these effects are at a lower magnitude. The training process of the classifier is naturally subject to fluctuations since usually a local minimum, not the global minimum, in the solution space is found. In general, this is not harmful, since the global minimum on the training data set does not necessarily represent the global minimum of the underlying distribution. The choice of the minimum depends heavily on the gradients during the training process. These are to some extent dependent on the starting conditions of the initialized weights of the algorithm.

To quantify the dependency of the weight initialization, multiple trainings with similar algorithm parameters, but different weight initializations on the same training data set are performed. Since the duration of the training procedure of the algorithm is currently of the order of two days, these tests have only been performed on 13 samples so far. In Figure B.3, the variation of the classifier shapes during the training process is shown. Variations of the classifier shape of different trainings are are visible. This indicates that the symmetry between B and $\overline{B}$ meson classification is not directly regularized by the loss function. Nevertheless, for the effective tagging efficiency, as defined in Section 5.4.1, stable values have been obtained. The mean $\bar{\mathcal{Q}} = 0.3532$ and a standard deviation of $\sigma_{\mathcal{Q}} = 0.0007$ is measured on a mono-generic test sample with $B^0 \to \nu\bar{\nu}$.

For measuring the symmetry between the classification process of both distributions, a χ^2 test is performed. Here, the classifier shape for B mesons only, and the mirrored distribution of the classifier shape for $\overline{B}$ mesons only, are compared. The classifier output is binned in two histograms $N_{B,i}$ and $N_{\overline{B},i}$ for B and $\overline{B}$ mesons respectively. Then a χ^2 value with

$$\chi^2 = \sum_{i=1}^{100} \frac{(N_{B,i} - N_{\overline{B},i})^2}{N_{B_i} + N_{\overline{B}_i}} \tag{.1}$$

is calculated. The measure for the classifier symmetry in each epoch during the training procedure is shown in Figure B.4. Multiple studies have been performed to regularize the symmetry. As additional regularization term in the loss function a χ^2 loss, the Kullback Leibler Divergence (used in Chapter 4) or the Wasserstein distance (used in Chapter 8) have been used. An addition of these penalties on the loss functions either had only a minor impact on the classifier symmetry or destabilized the training procedure. Further studies to stabilize this process are required, which are beyond the scope of this thesis.

Nevertheless, a purity transformation as in Equation (4.15) has an impact on the actual ability to symmetrize the classification process of the algorithm for the different B meson flavors. The relation between classifier symmetry and effective tagging efficiency is shown Figure B.5. Classifiers with and without a symmetry transformation are compared. A purity transformation seems to increase the the classifier symmetry in general.

For Belle II, the classifiers are trained and validated by a specialized subgroup and then uploaded to central database. These central weights are to be used in analyses.

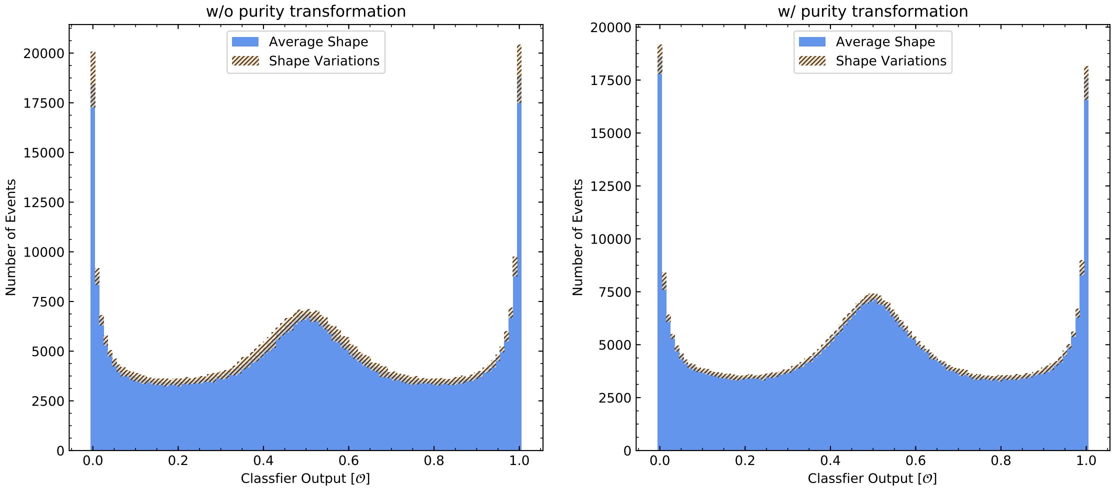

Figure B.3.: Variation of shapes of multiple classifier trainings with different seeds. The average and the minimum and maximum of all classifier trainings is shown. On the left hand side, results without a purity transformation are shown. A purity transformation is applied on the right hand side, and reduces the shape fluctuation of the classifier drastically.

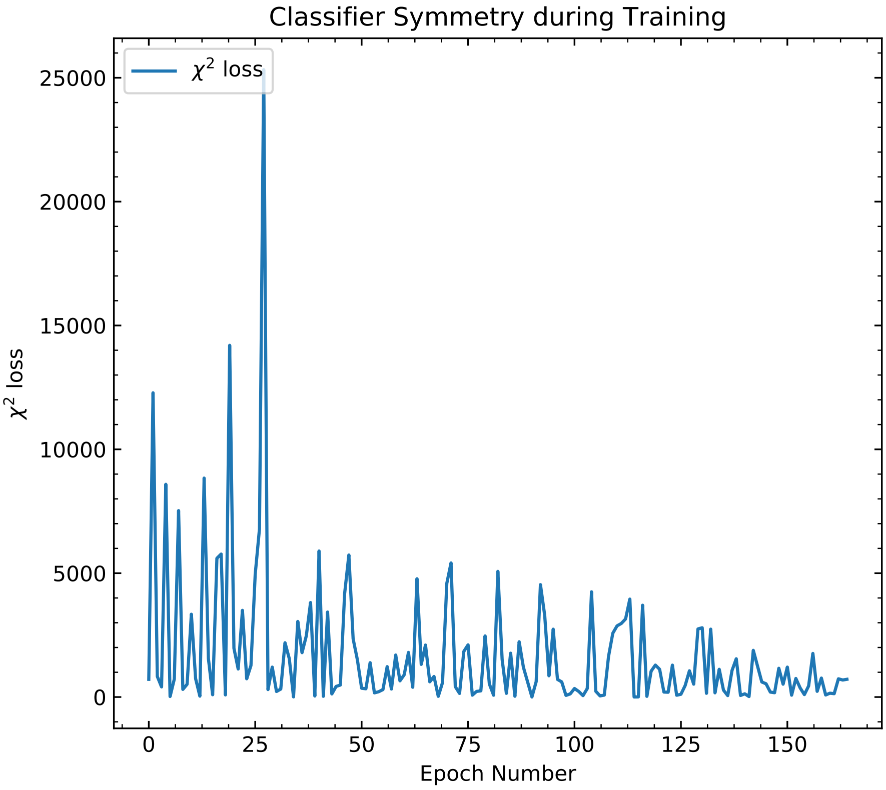

Figure B.4.: Monitoring a χ^2 loss on separate validation data set. The variations are decreasing with an increasing epoch number. Note, that the configuration of this network was altered during the studies and the number of hidden layers decreased to speed up the training speed.

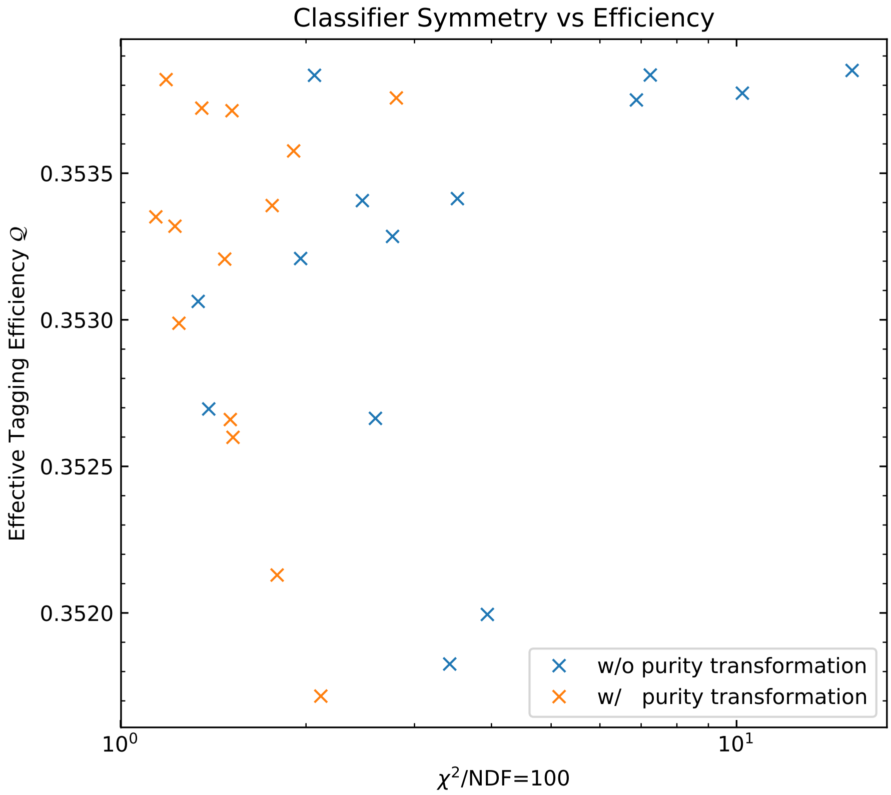

Figure B.5.: Comparison of symmetry and efficiency for multiple classifier trainings with different weight initializations. The χ^2 statistic for the classifier distributions with and without a purity transformation on an independent Monte Carlo data set are shown. With a lower χ^2 value the distributions of classifying B and $\overline{B}$ mesons can be distinguished less well, and therefore have a lower asymmetry. The number of degrees of freedom is determined by the number of bins.

For the uploaded weights, the shape differences of a selected classifier training are of only minor importance, once they are understood and calibrated. For the studies in this thesis, the classifier with the lowest value of the asymmetry has been picked. Nevertheless, it can be worthwhile to enforce a more symmetric classifier during the training procedure.

List of Figures

List of Tables

Bibliography

[1] A Abashian et al. „The belle detector“. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 479.1 (2002), pp. 117–232.

[2] AJ Bevan et al. „The physics of the B factories“. In: *The European Physical Journal C* 74.11 (2014), pp. 1–928.

[3] *KEKB schematic representation.* CC0. 2013. URL: https://de.wikipedia.org/wiki/KEKB#/media/Datei:KEKB.png (visited on 02/04/2020).

[4] T. Abe et al. „Belle II Technical Design Report“. In: (2010). arXiv: 1011.0352 [physics.ins-det].

[5] F. Abudinén et al. „Measurement of the integrated luminosity of the Phase 2 data of the Belle II experiment“. In: *Chin. Phys.* C41 (2020), p. 021001. DOI: 10.1088/1674-1137/44/2/021001. arXiv: 1910.05365 [hep-ex].

[6] I. Adachi et al. „Search for an Invisibly Decaying Z' Boson at Belle II in $e^+e^- \to \mu^+\mu^-(e^\pm\mu^\mp)$ Plus Missing Energy Final States“. In: (2019). submitted to Phys. Rev. Lett. arXiv: 1912.11276 [hep-ex].

[7] Toru Ijima. *SuperKEKB Long Term Luminosity.* 2020. URL: https://indico.belle2.org/event/1391/contributions/7363/ (visited on 04/13/2020).

[8] Kim Albertsson et al. „Machine Learning in High Energy Physics Community White Paper“. In: *J. Phys. Conf. Ser.* 1085.2 (2018), p. 022008. DOI: 10.1088/1742-6596/1085/2/022008. arXiv: 1807.02876 [physics.comp-ph].

[9] Sabrina Amrouche et al. „Track reconstruction at LHC as a collaborative data challenge use case with RAMP“. In: *EPJ Web Conf.* 150 (2017), 00015. 12 p. DOI: 10.1051/epjconf/201715000015. URL: https://cds.cern.ch/record/2280554.

[10] Rene Brun and Fons Rademakers. „ROOT—an object oriented data analysis framework“. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 389.1-2 (1997), pp. 81–86.

[11] T. Kuhr et al. „The Belle II Core Software“. In: *Comput. Softw. Big Sci.* 3.1 (2019), p. 1. DOI: 10.1007/s41781-018-0017-9. arXiv: 1809.04299 [physics.comp-ph].

[12] Andreas Moll. „The software framework of the Belle II experiment“. In: *Journal of Physics: Conference Series.* Vol. 331. 3. IOP Publishing. 2011, p. 032024.

[13] Sea Agostinelli et al. „GEANT4—a simulation toolkit“. In: *Nuclear instruments and methods in physics research section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506.3 (2003), pp. 250–303.

[14] Thomas Keck. „Machine learning algorithms for the Belle II experiment and their validation on Belle data“. PhD thesis. Karlsruher Institut für Technologie (KIT), 2017. 240 pp. DOI: 10.5445/IR/1000078149.

[15] Thomas Keck. „FastBDT: a speed-optimized multivariate classification algorithm for the Belle II experiment“. In: *Computing and Software for Big Science* 1.1 (2017), p. 2.

[16] Thomas Keck et al. „B2BII: Data Conversion from Belle to Belle II“. In: *Comput. Softw. Big Sci.* 2.1 (2018), p. 9. DOI: 10.1007/s41781-018-0016-x. arXiv: 1810.00019 [hep-ex].

[17] Markus Röhrken. „Time-Dependent CP Violation Measurements in Neutral B Meson to Double-Charm Decays at the Japanese Belle Experiment“. PhD thesis. Karlsruhe U., 2012. DOI: 10.5445/IR/1000028856.

[18] Geoffrey C Fox and Stephen Wolfram. „Observables for the analysis of event shapes in e+ e- annihilation and other processes“. In: *Physical Review Letters* 41.23 (1978), p. 1581.

[19] T. Keck et al. „The Full Event Interpretation“. In: *Comput. Softw. Big Sci.* 3.1 (2019), p. 6. DOI: 10.1007/s41781-019-0021-8. arXiv: 1807.08680 [hep-ex].

[20] Mark Thomson. *Modern particle physics.* Cambridge University Press, 2013.

[21] Michael Peskin. *An introduction to quantum field theory.* Westview Press, 1995.

[22] *Standard Model of Particle Physics.* CC0. 2017. URL: https://commons.wikimedia.org/wiki/File:Standard_Model_of_Elementary_Particles.svg (visited on 04/12/2020).

[23] G. Aad et al. „Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC“. In: *Physics Letters B* 716.1 (Sept. 2012), pp. 1–29. ISSN: 0370-2693. DOI: 10.1016/j.physletb.2012.08.020. URL: http://dx.doi.org/10.1016/j.physletb.2012.08.020.

[24] Serguei Chatrchyan et al. „Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC“. In: *Physics Letters B* 716.1 (2012), pp. 30–61.

[25] JP Lees et al. „Measurement of an excess of B$^-$→ D (*) τ- ν^- τ decays and implications for charged Higgs bosons“. In: *Physical Review D* 88.7 (2013), p. 072012.

[26] Makoto Kobayashi and Toshihide Maskawa. „CP Violation in the Renormalizable Theory of Weak Interaction“. In: *Prog. Theor. Phys.* 49 (1973), pp. 652–657. DOI: 10.1143/PTP.49.652.

[27] M. Tanabashi et al. „Review of Particle Physics“. In: *Phys. Rev. D* 98 (3 Aug. 2018), p. 030001. DOI: 10.1103/PhysRevD.98.030001. URL: https://link.aps.org/doi/10.1103/PhysRevD.98.030001.

[28] Lincoln Wolfenstein. „Parametrization of the Kobayashi-Maskawa matrix“. In: *Physical Review Letters* 51.21 (1983), p. 1945.

[29] J. Charles et al. In: *Eur. Phys. J.* C41 (2005). updated results and plots available at: `http://ckmfitter.in2p3.fr`, pp. 1–131. DOI: `10.1140/epjc/s2005-02169-1`. arXiv: `hep-ph/0406184 [hep-ph]`.

[30] Andrzej J Buras, Markus E Lautenbacher, and Gaby Ostermaier. „Waiting for the top quark mass, K+→ π+ $\nu\nu^{-}$, B s 0-B¯ s 0 mixing, and CP asymmetries in B decays“. In: *Physical Review D* 50.5 (1994), p. 3433.

[31] J. H. Christenson et al. „Evidence for the 2π Decay of the K_2^0 Meson“. In: *Phys. Rev. Lett.* 13 (4 July 1964), pp. 138–140. DOI: `10.1103/PhysRevLett.13.138`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.13.138`.

[32] H Albrecht et al. „Observation of B0-anti-B0 Mixing“. In: *Phys. Lett.* 192.DESY-87-029 (1987), pp. 245–252.

[33] Thomas Kuhr. „Flavor physics at the Tevatron“. In: *Springer Tracts Mod. Phys.* 249 (2013), pp. 1–161.

[34] R. Aaij et al. „Observation of CP Violation in Charm Decays“. In: *Phys. Rev. Lett.* 122 (21 May 2019), p. 211803. DOI: `10.1103/PhysRevLett.122.211803`. URL: `https://link.aps.org/doi/10.1103/PhysRevLett.122.211803`.

[35] Belle Collaboration and K. Abe. „Observation of Mixing-induced CP Violation in the Neutral B-meson System“. In: (2002), p. 25. DOI: `10.1103/PhysRevD.66.032007`. arXiv: `0202027 [hep-ex]`. URL: `http://arxiv.org/abs/hep-ex/0202027`.

[36] Thomas E Browder and Klaus Honscheid. „B mesons“. In: *Progress in Particle and Nuclear Physics* 35 (1995), pp. 81–219.

[37] ed. Harrison P.F., ed. Quinn Helen R., and /SLAC. „The BABAR Physics Book: Physics at an Asymmetric B Factory“. In: (May 2010). DOI: `10.2172/979931`.

[38] Jochen Gemmler. „Study of B Meson Flavor Tagging with Deep Neural Networks at Belle and Belle II“. Karlsruhe Institute of Technology (KIT), Masterarbeit, 2016. MS. Karlsruhe Institute of Technology (KIT), 2016. URL: `https://ekp-invenio.physik.uni-karlsruhe.de/record/48849`.

[39] H. Kakuno et al. „Neutral B flavor tagging for the measurement of mixing-induced CP violation at Belle“. In: *Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 533.3 (2004), pp. 516–531. ISSN: 01689002. DOI: `10.1016/j.nima.2004.06.159`. arXiv: `0403022 [hep-ex]`.

[40] Moritz Gelb. „Neutral B Meson Flavor Tagging for Belle II“. MA thesis. Karlsruhe Institute of Technology (KIT), 2015.

[41] Fernando Abudinén. „Development of a B^0 flavor tagger and performance study of a novel time-dependent CP analysis of the decay $B^0 \to \pi^0\pi^0$ at Belle II“. PhD thesis. 2018.

[42] Christopher M Bishop. *Pattern recognition and machine learning.* springer, 2006.

[43] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning.* MIT press, 2016.

[44] Andrew W Senior et al. „Improved protein structure prediction using potentials from deep learning“. In: *Nature* (2020), pp. 1–5.

[45] Victor Bapst et al. „Unveiling the predictive power of static structure in glassy systems“. In: *Nature Physics* 16.4 (2020), pp. 448–454.

[46] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. „Deep learning“. In: *nature* 521.7553 (2015), pp. 436–444.

[47] Yoshua Bengio, Aaron Courville, and Pascal Vincent. „Representation Learning: A Review and New Perspectives“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.8 (2012), pp. 1798–1828. ISSN: 1939-3539. DOI: `10.1109/TPAMI.2013.50`. arXiv: `1206.5538`.

[48] Pierre Baldi, Peter Sadowski, and Daniel Whiteson. „Searching for exotic particles in high-energy physics with deep learning“. In: *Nature communications* 5.1 (2014), pp. 1–9.

[49] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: `https://www.tensorflow.org/`.

[50] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. „Practical bayesian optimization of machine learning algorithms“. In: *Advances in neural information processing systems*. 2012, pp. 2951–2959.

[51] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. „Learning representations by back-propagating errors“. In: *nature* 323.6088 (1986), pp. 533–536.

[52] Laurens van der Maaten and Geoffrey Hinton. „Visualizing data using t-SNE“. In: *Journal of machine learning research* 9.Nov (2008), pp. 2579–2605.

[53] Erik Bernhardsson, Elias Freider, et al. *Luigi - A python module that helps to build complex pipelines for batch jobs*. https://github.com/spotify/luigi. 2012-2020.

[54] Fernando Pérez and Brian E. Granger. „IPython: a System for Interactive Scientific Computing“. In: *Computing in Science and Engineering* 9.3 (May 2007), pp. 21–29. ISSN: 1521-9615. DOI: `10.1109/MCSE.2007.53`. URL: `https://ipython.org`.

[55] Wes McKinney. „pandas: a foundational Python library for data analysis and statistics“. In: *Python for High Performance and Scientific Computing* 14 (2011).

[56] The HDF Group. *Hierarchical Data Format, version 5*. http://www.hdfgroup.org/HDF5/. 1997-2020.

[57] David J Lange. „The EvtGen particle decay simulation package“. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 462.1-2 (2001), pp. 152–155.

[58] René Brun et al. *GEANT 3: user's guide Geant 3.10, Geant 3.11*. Tech. rep. CERN, 1987.

[59] L. Hinz, Jacoby C., and J. Wicht. „Lepton efficiency and systematic error for experiments 21 to 27“. In: *Internal Belle Note* 777 (2004).

[60] L. Hinz. „Lepton ID efficiency correction and systematic error“. In: *Internal Belle Note* 954 (2006).

[61] S. Nishida. „Study of Kaon and Pion Identification Using Inclusive D^* Sample“. In: *Internal Belle Note* 779 (2004).

[62] P. Koppenburg. „A Measurement of the Track Finding Efficiency Using Partially Reconstructed D^* Decays“. In: *Internal Belle Note* 621 (2003).

[63] Glen Cowan. *Statistical data analysis*. Oxford university press, 1998.

[64] Alfio Lazzaro and Lorenzo Moneta. „MINUIT package parallelization and applications using the RooFit package“. In: *Journal of Physics: Conference Series*. Vol. 219. 4. IOP Publishing. 2010, p. 042044.

[65] Wouter Verkerke and David Kirkby. „The RooFit toolkit for data modeling“. In: *Statistical Problems in Particle Physics, Astrophysics and Cosmology*. World Scientific, 2006, pp. 186–189.

[66] Simon Wehle. *PyrooFit - A fit framework for python on top of ROOT.RooFit.* https://github.com/simonUU/PyrooFit. 2018-2019.

[67] Eric O Lebigot. *Uncertainties: a Python package for calculations with uncertainties*. http://pythonhosted.org/uncertainties. 2010-2017.

[68] Daniel Guest et al. „Jet flavor classification in high-energy physics with deep neural networks“. In: *Physical Review D* 94.11 (2016), p. 112002.

[69] Fernando Abudinén et al. *Flavor tagging with Belle II data*. Oct. 2019. URL: `https://indico.belle2.org/event/971/contributions/4838/attachments/2579/3880/2019_10_22B2GM.pdf`.

[70] Muriel Pivk and Francois R Le Diberder. „Plots: A statistical tool to unfold data distributions“. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 555.1-2 (2005), pp. 356–369.

[71] S Agostinelli et al. „Geant4—a simulation toolkit“. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506.3 (2003), pp. 250–303. ISSN: 0168-9002. DOI: `http://dx.doi.org/10.1016/S0168-9002(03)01368-8`. URL: `http://www.sciencedirect.com/science/article/pii/S0168900203013688`.

[72] Michela Paganini, Luke de Oliveira, and Benjamin Nachman. „Accelerating science with generative adversarial networks: an application to 3D particle showers in multilayer calorimeters“. In: *Physical review letters* 120.4 (2018), p. 042003.

[73] Yann LeCun et al. „Handwritten digit recognition with a back-propagation network“. In: *Advances in neural information processing systems*. 1990, pp. 396–404.

[74] Ian Goodfellow et al. „Generative adversarial nets“. In: *Advances in neural information processing systems*. 2014, pp. 2672–2680.

[75] Martin Arjovsky, Soumith Chintala, and Léon Bottou. „Wasserstein gan“. In: *arXiv preprint arXiv:1701.07875* (2017).

[76] Ishaan Gulrajani et al. „Improved training of wasserstein gans“. In: *Advances in neural information processing systems*. 2017, pp. 5767–5777.

[77] Martin Erdmann, Jonas Glombitza, and Thorben Quast. „Precise simulation of electromagnetic calorimeter showers using a Wasserstein Generative Adversarial Network“. In: *Computing and Software for Big Science* 3.1 (2019), p. 4.

[78] Martin Erdmann et al. „Generating and refining particle detector simulations using the Wasserstein distance in adversarial networks“. In: *Computing and Software for Big Science* 2.1 (2018), p. 4.

[79] Ashish Vaswani et al. „Attention is all you need“. In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[80] *Abudinén, Fernando*. personal communication. May 2, 2019.

Danksagung

Ich bedanke mich bei Herrn Prof. Michael Feindt für die Ermöglichung dieser Arbeit in diesem spannenden Themengebiet und die hilfreichen Ratschläge.

Ganz herzlich bedanke ich mich bei Herrn Prof. Florian Bernlochner, für die großartige Unterstützung und interessanten fachlichen Diskussionen, die wesentlich zum Gelingen dieser Arbeit beigetragen haben.

Zudem möchte ich mich bei Herrn Martin Heck bedanken, der mir das Thema Deep Learning näher gebracht hat.

Besonderen Dank gilt auch Herrn Pablo Goldenzweig für die vielen hilfreichen Kommentare, Empfehlungen und anregenden physikalischen Diskussionen.

Ich bedanke mich bei allen Kollegen am Institut, namentlich möchte ich hier Herr Thomas Keck für seine unerschöpfliche Inspiration und hilfreichen Ratschläge, sowie Herr Felix Metzner für die zahlreichen spannenden fachlichen Diskussionen erwähnen.

Zudem bedanke ich mich bei allen Doktoranden, die die IT-Infrastruktur am Institut verwalten, teilweise auch in ihrer Freizeit, und es somit ermöglichen, eine flexible und funktionelle Basis für das Computing bereitzustellen.

Des Weiteren bedanke ich mich bei Herrn Pablo Goldenzweig, Herrn Felix Metzner, Herrn Michael Eliachevitch und Herrn Karsten Wolff für das Korrekturlesen dieser Arbeit.

Ganz besonders möchte ich mich auch bei meiner Familie für ihre unerschöpfliche Unterstützung bedanken.